SANDIA REPORT

SAND2017-2779

Unlimited Release

Printed March 2017

Spatial and Temporal Analysis of Bias HAST System Temperature

Kent B. Pfeifer
Clint T. Furrer III
Paul A. Sandoval
Stephen E. Garrett
Nathaniel B. Pfeifer

Prepared by Sandia National Laboratories

Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@osti.gov
Online ordering: http://www.osti.gov/scitech

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5301 Shawnee Rd
Alexandria, VA 22312

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.gov
Online order: http://www.ntis.gov/search

SAND2017-2779
Unlimited Release
Printed March 2017

Spatial and Temporal Analysis of Bias HAST System Temperature

Kent B. Pfeifer[a]
Clint T. Furrer III[b]
Paul A. Sandoval[c]
Stephen E. Garrett[b]
Nathaniel B. Pfeifer[a]

[a]Nano and Micro Sensors, 8643
[b]Microsystem Integration, 1718
[c]Packaging/Assembly Operations, 1718-1
Sandia National Laboratories
P.O. Box 5800
Albuquerque, New Mexico 87185-MS1425

Abstract

High-reliability components for high-consequence systems require detailed testing of operation after having undergone highly accelerated stress testing (HAST) under unusual conditions of high-temperature and humidity. This paper describes the design and operation of a system called "Wormwood" that is a highly multiplexed temperature measurement system that is designed to operate under HAST conditions to allow measurement of the temperature as a function of time and position in a HAST chamber. HAST chambers have single-point temperature measurements that can be traceable to NIST standards. The objective of these "Wormwood" measurements is to verify the uniformity and stability of the remaining volume of the HAST chamber with respect to the single traceable standard.

CONTENTS

FIGURES

NOMENCLATURE

HAST	Highly Accelerated Stress Testing
BHAST	Biased Highly Accelerated Stress Testing
RH	Relative Humidity
RTD	Resistance Temperature Detector
PSL	Sandia National Laboratories Primary Standards Laboratory
NIST	National Institute of Standards and Technology
Wormwood	Something bitter or grievous[1]; Ukrainian translation is Chernobyl[2]

1. INTRODUCTION

High-reliability components for high-consequence systems must be tested after experiencing environmental stresses to verify that function is not compromised due to corrosion or other

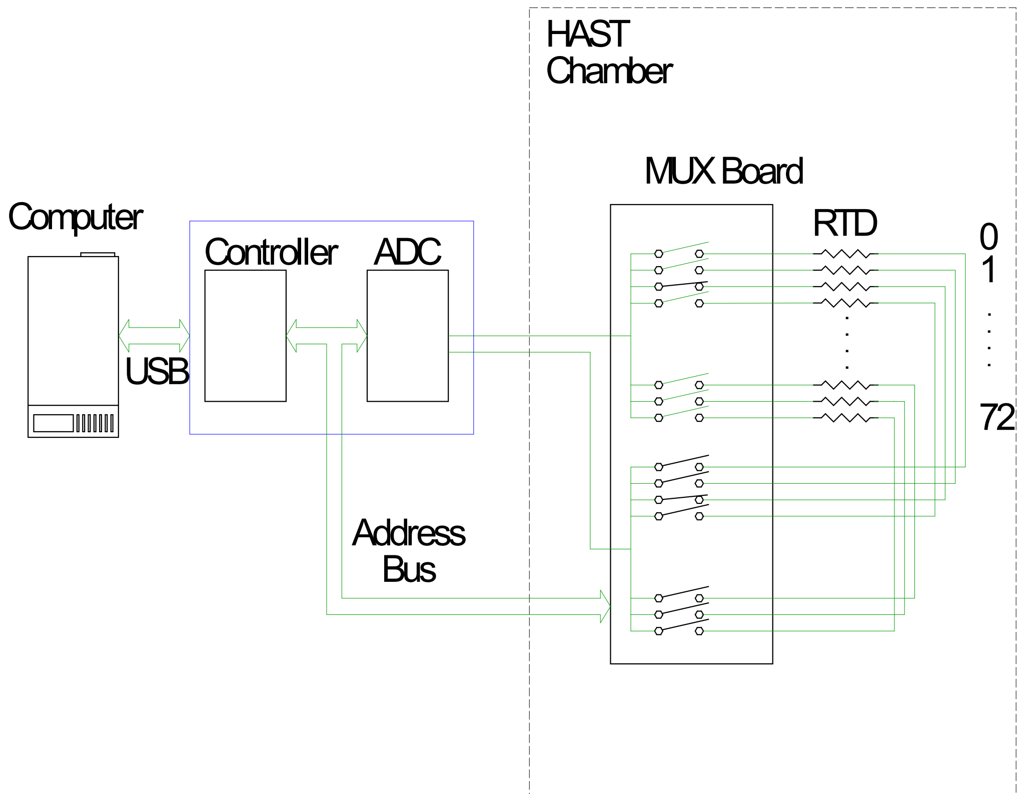


Figure 1: Schematic diagram of the Wormwood system illustrating the various components and their physical relationship to the HAST chamber. The design is guided by minimizing the active electronic hardware that is necessarily placed in the HAST chamber. Figure shows the control, ADC, and multiplexer boards as well as the 72 RTDs for measurement.

effects.[3,4] For systems with high-reliability requirements, highly accelerated stress testing (HAST) is done to components to estimate the functional life-span of the component under normal environmental conditions. Post HAST testing for electrical function and post-mortem testing of failed parts using advanced failure analysis techniques allows designers to provide realistic, physics-based estimates of component reliability and the ability to extrapolate system reliability from the estimated life-spans of the components.

HAST measurement equipment must be able to provide high-temperature (>130C) environments with high relative humidity levels (>85% RH). This requires HAST chambers that must be capable of withstanding high pressures to achieve these environments. For example, the pressure of 85% relative humidity (RH) at 130C is about 140 kPa which is about 1.4 atmospheres. In addition, the chambers are rated for as high as 700 kPa (~7 atm). As a result, the interior of a

HAST chamber can afford little electrical access for monitoring the interior and the limited access provided is typically wired for biased HAST (BHAST) measurements limiting the number of temperature measurement nodes that can be realized.

Since there is little access for measurement, system control is accomplished using a heating element in the bottom of the chamber and a single-point thermocouple in the top of the chamber and assumes that the rest of the system is uniformly controlled under quasi-static thermodynamic conditions. However, evidence from testing, such as melted component handling equipment and lots with melted solder balls, implies that the uniformity of the chamber is not ideal. To confirm that individual parts are not likely to have experienced outlier temperatures due to their spatial location during the test, our system was designed to make a measurement of the temperature at a

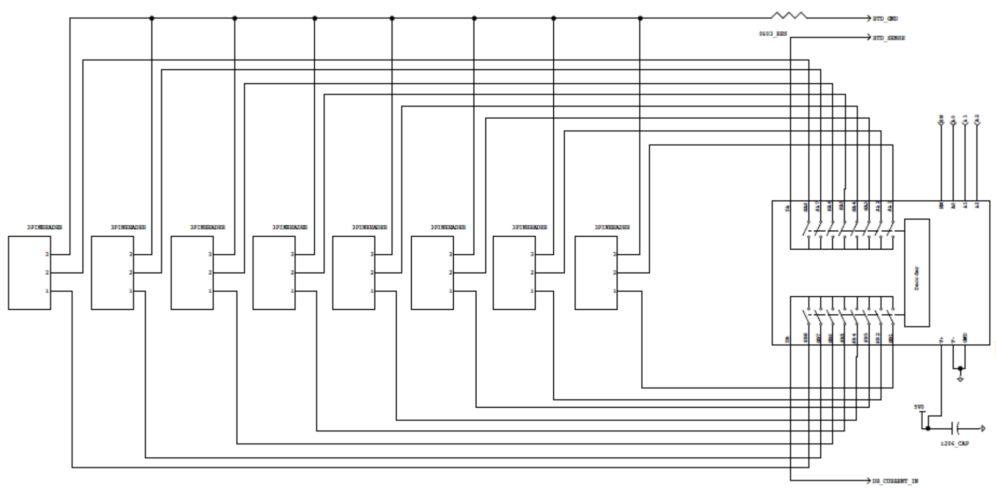


Figure 2: Schematic diagram of one multiplexer and its associated 8 RTDs. The RTDs are connected into the sockets on the left of the multiplexer. The multiplexer switches both the incoming current and the voltage across the RTD. The multiplexer is addressed using the three address lines and the chip select line illustrated in the schematic.

specific location within the HAST chamber.

Thus, the “Wormwood” system was constructed to provide both a a spatial and temporal map of the temperature at 72 individual points in the chamber during HAST operation. This requires that the measurement system itself be constructed of circuit materials and components capable of withstanding high-temperatures and pressures under the additional stress provided by operational bias. Essentially, the system must function under the HAST conditions while providing immunity to the deleterious electrochemical effects that occur when electronic components are in operation under these conditions. These conditions are known as bias HAST (BHAST).

1.1. System Design

The design of the system is dictated by the limitations of electronic components under high temperature and pressure conditions. For example, most commercial specification components are temperature rated for operation at below 85ºC with military specification components rated at below 125ºC. In both cases, the temperature requirements of the HAST test exceed the limitations of the components. As a result, we designed the system to have the majority of the components located outside of the HAST chamber and use solid-state relays to bring the signal from the RTD to the outside world via a high-temperature, high-pressure rated multiplexer (Honeywell P/N HT507[a]) which is specified to operate over an extended temperature range of -55ºC to +225ºC. The schematic of a single multiplexer section of the system is given in Figure 2. In addition, the circuit boards were fabricated by *HAST Solutions*[b] to our specifications but

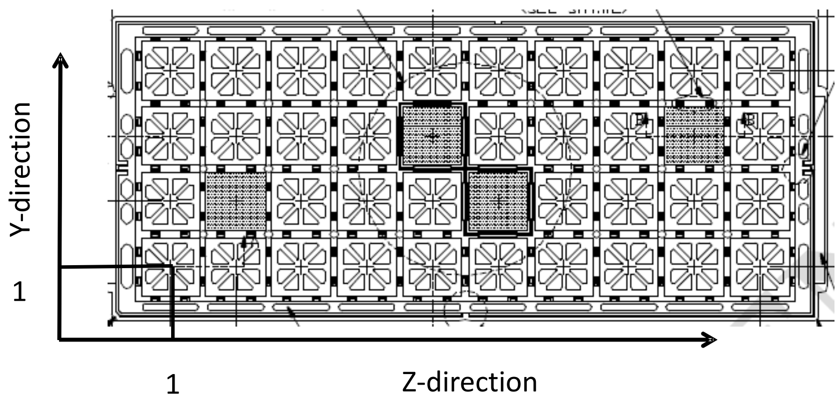


Figure 3: Diagram showing the test component handling fixture. Each part occupies a square on the fixture and the Wormwood system is designed to measure the temperature at each location where position (1,1) is shown and the positive z- and y-directions are shown.

using materials and layout techniques that have been proven to survive for thousands of hours of HAST testing. In addition, this architecture reduces the number of connections from the inside of the HAST chamber to the outside of the HAST chamber and can be accommodated by the sealed access ports of the HAST chamber. The general architecture of the system is illustrated in Figure 1 which shows the RTD and multiplexer board in the HAST chamber with the controller and ADC located outside in a normal operating environment. The data is digitized and sent to the computer via USB bus for storage as shown.

[a] Honeywell, 12001 Highway 55, Plymouth, MN 55441, Tel: 800-323-8295, www.honeywell.com/hightemp.

[b] HAST Solutions, Inc., 1800 Green Hills Rd Suite 107, Scotts Valley, CA 95066.

1.1.1. Hardware

The sensor is an array of resistance temperature detectors (RTD) that map directly onto the position of the parts under test in the handling fixture. This is illustrated in Figure 3 which shows a diagram of the component holder and the coordinates for locating the parts in the carrier. Mechanically, the circuit board for each chip carrier is located parallel to the component holder and both the circuit board and the component holder ride vertically in the HAST chamber. Each

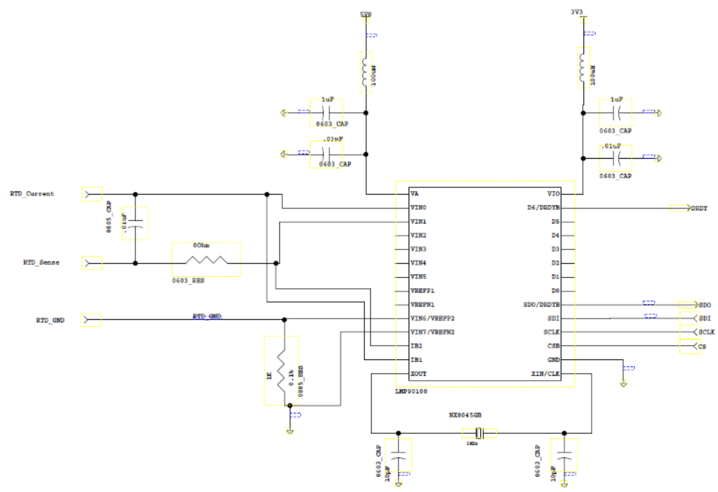


Figure 4: Schematic diagram of the ADC section of the system. The RTD current, sense, and ground lines originate on the ADC chip and are shown at the left-hand side of the figure. Communication with the chip is accomplished via the SPI interface illustrated on the right-hand side of the schematic.

RTD from the board is mapped over to a unit cell of the component carrier such the measurement of temperature represents the location of an individual test component. The board is equipped with the previously mentioned multiplexer chip such that 8 channels are controlled by each chip with three multiplexer chips on each board for a total of 24 temperature channels on each board as is illustrated in Figure 2.

The ADC/controller board (Figure 1) is located outside the HAST chamber and is based around a LMP90100[c] ADC chip with 24-bits of resolution. The schematic of the ADC circuit is shown in Figure 4 which illustrates both the interface with the RDT and the digital communication with the component. Examination of Figure 4 shows that the ADC is capable of supporting only one channel of RTD. Thus, a series of reed relays (Pickering 1030[d]) are placed between the

[c] Texas Instruments, Post Office Box 655303, Dallas, Texas 75265.

multiplexer board and the ADC to bring the signal from the RTD to the ADC with minimum distortion. Reed relays were chosen due to their small series resistance (<150mΩ) and provide little distortion to the temperature signal.

The system functions by sourcing a constant current from the LMP90100 through the RTD and

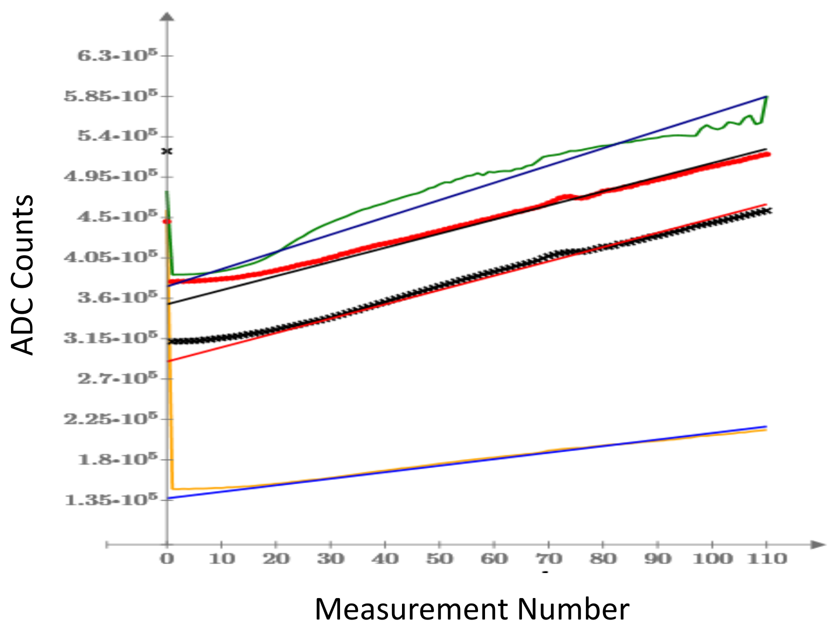


Figure 5: Example data illustrating the fit of the temperature ramp using the MathCAD program to find the slope and the y-intercept for each channel of the Wormwood system. The data are plotted as the green, red, black data points, and orange traces. The fit is calculated from Eq. * MERGEFORMAT (1.2) and plotted as the solid blue, solid black, solid red, and solid blue traces (labeled from top to bottom).

measuring the voltage drop across a reference resistor. Since the resistance of the RTD is a primary measurement of the temperature of the RTD fabrication material, the voltage drop is a direct measurement of the absolute temperature of the RTD. Thus, the system should eliminate the need for an individual calibration of each RTD in the system. There is a problem, however, that arose as a result of the use of the HT507 in the system. As the series resistance of the solid-state relay is around 400 Ω compared to the total resistance of the RTD of 100 Ω, it became necessary to do a two point calibration of each RTD element to compensate. This will be discussed in detail in a later section, but is conducted as part of the HAST operation and does not substantially complicate Wormwood operation. Normally, this would be bad design practice to include this series resistance in the signal stream; however, the other limitations of operation in a HAST environment left little option but to use the HT507 and include the calibration in the data.

[d] Pickering Interfaces Inc., 2900 NW Vine Street, Suite D, Grants Pass, OR 97526, United States, Telephone: +1 541-471-0700.

The overarching system control was accomplished using a TI MSP430[c] programmed to address the RTDs and to read the ADC via the interface and send it to the computer for logging. This was implemented on a custom logic board whose schematic is shown in Appendix A and whose firmware is included in Appendix B. The main control code was written in Python and performs the relatively simple function of selecting an RTD by control of the addressing of the reed relays and the HT507 multiplexers and then initiating the measurement and storing that measurement. The data is stored as a text (.txt) file that can be imported into an Excel spreadsheet. The data is in the form of columns being the temperature on a particular thermocouple and the rows corresponding to the time of the measurement during the HAST run. In addition, to the data from the Wormwood board, we also employ the logged data from the HAST chamber which gives the temperature at the control thermocouple as a function of time.

1.1.1. Data Analysis

Data are received from the Wormwood system in the form of a temperature measurement in degrees Celsius; however, they must be reconverted to ADC counts due to the parasitic resistance issues discussed above. This is done by using a MathCAD[e] program to implement the following correction equation found from the TI MSP430 data sheet[c]:

$$ADC\_Counts = \left(Data + 257.01\right)1632 \quad \text{* MERGEFORMAT (1.1)}$$

The data are a matrix of measurements that correspond to the sensor channel for each column and the time of measurement for each matrix row. Since we record the heating profile of the HAST chamber as well as the steady-state conditions, we are able to use the logging data from the HAST chamber to realize a two-point calibration of the HAST chamber that is traceable to the PSL calibrated thermocouple on the HAST chamber. By observation, the data are parsed into sections from the beginning at the initial temperature of the HAST chamber to the data measurements occurring at the operation point of the HAST chamber (T_{final} usually 130C). This data is then fit to a linear function of the standard form:

$$ADC\_Counts = m \times measurement\_number + offset \quad \text{* MERGEFORMAT (1.2)}$$

Where *ADC_Counts* is the variable name representing the data fit in counts, *m* is the calculated fit of the slope of the temperature profile, and *offset* is the y-intercept of the data. In this case, the value *offset* is the initial ADC measurement at the beginning temperature (T_1 usually room temperature) of the measurement. An example of this analysis is shown in Figure 5 and illustrates an excellent straight-line fit of the ramp data for several example channels. Having this fit data allows an analytical calculation of the "knee" point of the HAST temperature profile. The "knee" is the point at which the temperature profile changes from a heating profile to a iso-thermal temperature profile.

---

[e] *PTC Corporate Headquarters 140 Kendrick Street,Needham, MA 02494, USA.*

Having accomplished this, we can convert the ADC count data to traceable temperature for each RTD channel. The data in each column is a series of raw ADC measurements at each measurement interval. Again, each column represents an individual sensor. Since the data are in raw ADC counts, they need to be transformed into corresponding temperatures. This is done by

3, 2, 1

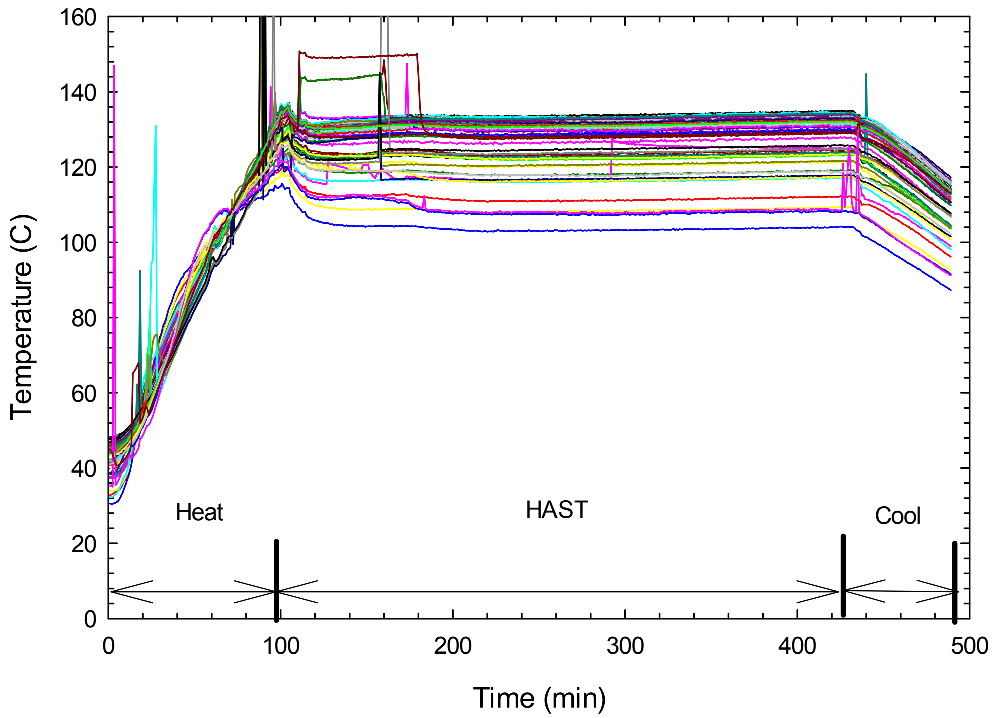


Figure 6: Figure showing representative data from the Wormwood system during a HAST run. Data has been converted to temperature via Eq. * MERGEFORMAT (1.6). The HAST chamber is set to 130C using the PSL traceable thermocouple and the internal control system. Data are stable during the HAST section of the run to less than ±5C.

way of proposing a linear transformation of the data based on two parameters. For the sake of our discussion, these will be α and β and are defined as follows:[5]

$$\begin{aligned} T_N &= \alpha_n \; ADC\_Counts_{n,N} + \beta_n \\ T_1 &= \alpha_n \; ADC\_Counts_{n,1} + \beta_n \end{aligned} \qquad \text{* MERGEFORMAT (1.3)}$$

Where n is the RTD channel number, N is the data point of the temperature measurement, and T is the resultant temperature in degrees Celsius for RTD channel n. In matrix form, Eq. * MERGEFORMAT (1.3) is written as:

$$\begin{bmatrix} T_N \\ T_1 \end{bmatrix} = \begin{bmatrix} ADC\_Counts_{n,N} & 1 \\ ADC\_Counts_{n,1} & 1 \end{bmatrix} \begin{bmatrix} \alpha_n \\ \beta_n \end{bmatrix} \qquad \text{* MERGEFORMAT (1.4)}$$

Inversion of the matrix leads to values of α and β for each RTD channel as follows:

$$\begin{bmatrix} \alpha_n \\ \beta_n \end{bmatrix} = \begin{bmatrix} ADC\_Counts_{n,N} & 1 \\ ADC\_Counts_{n,1} & 1 \end{bmatrix}^{-1} \begin{bmatrix} T_N \\ T_1 \end{bmatrix} \quad \text{* MERGEFORMAT (1.5)}$$

Thus in Eq. * MERGEFORMAT (1.5), the beginning and the final value of temperature (in Celsius) in the HAST chamber are inserted in the column matrix for temperature on the right-hand side of the equation. The values for the initial measurement (T_1) and N^{th} measurement at the "knee" are placed in the 1x2 matrix and are found from the fit equation (Eq. * MERGEFORMAT (1.2)). The result is a set of α and β parameters for each column to correct to

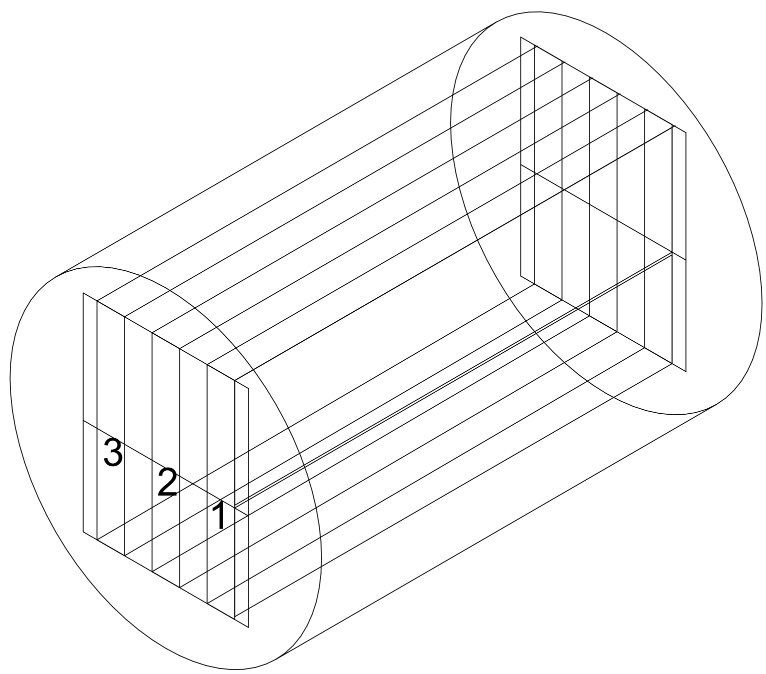


Figure 7: Diagram of the placement of the Wormwood board in the cylindrical HAST chamber. This is for a full 72 channel array of RTD sensors placed as illustrated in Figure 3.

temperature in degrees Celsius for each additional measurement data point from the Wormwood system.

The final step is to then take all of the measured data and convert to temperature in degrees Celsius. This is done as follows:

$$T_i = \alpha_n \ ADC\_Counts_{i,n} + \beta_n \qquad \text{* MERGEFORMAT (1.6)}$$

Eq. * MERGEFORMAT (1.6) transforms an RTD measurement in raw counts to its equivalent temperature in degrees Celsius incorporating the two-point calibration data. The rest of the data from the Wormwood is then corrected using Eq. * MERGEFORMAT (1.6) and plotted.

A representative data plot is shown in Figure 6 and illustrates a measurement in a HAST chamber using the system. Due to the extreme conditions, several of the runs have noise spikes and dead channels that must be edited prior to fitting. The data in Figure 6 is a HAST run starting at room temperature and ramping to 130C. The HAST system is left at 130C for ~500 min and the data is analyzed. This data is for a single board placed in the center of the chamber and measured. For most of the testing we employed three boards placed as shown in

17	10		24		12	8	7		6
	X	18		23	X	13		16	5
19		21	9	X		2	14	X	15
20			22	11	1		3		4

Tray 1

41	34		48		36	32	31		30
	x	42		47	x	37		40	29
43		45	33	x		26	38	x	39
44			46	35	25		27		28

Tray 2

65	58		72		60	56	55		54
	x	66		71	x	61		64	53
67		69	57	x		50	62	x	63
68			70	59	49		51		52

Tray 3

Figure 8: Map of sensor placement for 72 channel array. The location of each sensor corresponds to a cell in the component carrier shown in Figure 3. The red numbers are RTDs that have suspect performance or have experienced total failure due to the environment.

Figure 7.

For the full 72 channel array of sensors, a placement drawing is shown in Figure 8.

2. RESULTS

Data acquired from the Wormwood system demonstrate not only the temporal stability of the HAST system but also demonstrate the spatial uniformity of the HAST system. An example of this data is shown in Figure 9 which is a screen shot from an animation of the temperature for each board as a function of position. The software to accomplish this animation is written for MATLab[f] and is printed in Appendix C. The position data is acquired from the physical dimensions of the part carrier (Figure 3) and the RTD location map (Figure 8). The data show that the parts are heated but there is some non-uniformity in the temperature control over the board. For example, the lower board has a cool spot that is near position (1,1). This non-uniformity is accounted for in the operational specifications and documented uncertainty of the

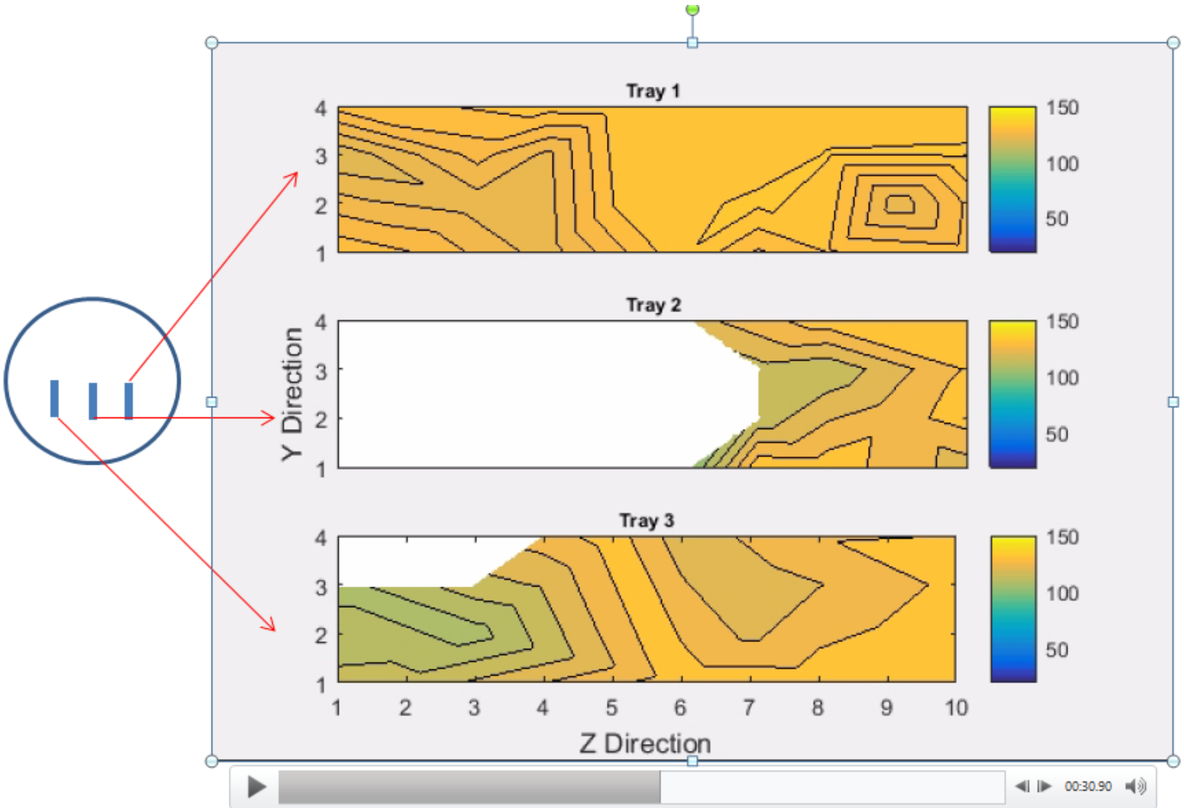


Figure 9: Animated plot of the temperature as a function of position in the chamber. The rectangle represents the spatial location of the sensor formed into a thermal contour map as a function of time. The white areas are places where we had failed channels and were not included in the contour mapping. The illustration at the left is the location of the board looking into the cylindrical HAST chamber.

---

[f] Math Works, 1 Apple Hill Drive, Natick, MA 01760-2098 UNITED STATES.

measurement. In addition, the white areas are generated from regions that have noisy or failed RTD data and are not included in the calculation.

Temporal data (Figure 6) illustrates the variation that can be expected from the HAST chamber. In this example, the temperature is set to 130C on the HAST thermocouple and the resulting measurement shows an average of 124 ±8C. This implies excellent agreement with the setting and establishes the limits of uncertainty in the HAST chamber.

Similar data were acquired for each of the HAST chambers and a representative sample is shown in Figure 10. In order to establish a bound on the uncertainty of the chambers as a function of space and time, the data acquired after equilibrium at around 200 min into the HAST runs for the individual values were plotted. The data in Figure 10 illustrate the variation from run-to-run in a single HAST chamber including the ±3σ and ±5σ boundaries that constitute the 95% and 99% confidence limits respectively.[6] Additionally, there is variation from run-to-run between HAST chambers when each RTD channel is placed in a similar location within the chamber. The standard deviation over the entire data set is found to be approximately ±10C with outliers as much as ±30C for individual channels.

3. CONCLUSIONS

The Wormwood system was constructed to facilitate measurement of the spatial and temporal variation encountered during HAST of plastic ball grid arrays at Sandia National Laboratories.

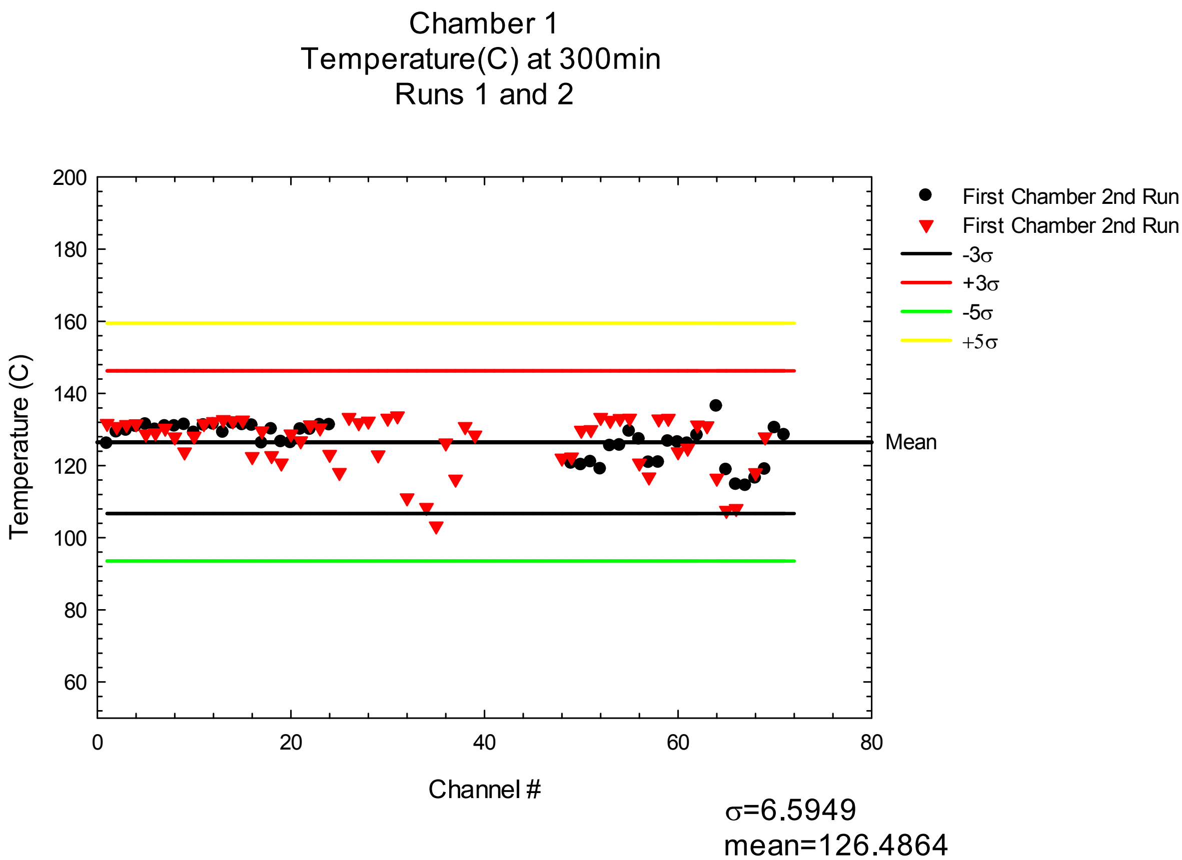


Figure 10: Plot of the data acquired from chamber 1 plotted as a function of channel number. Note that the channels between 24 and 47 have issues due to the severity of the environment and some have been edited out. Plot shows the mean of all channels sampled as well as the ±3σ and ±5σ lines around the data. Data illustrate the spatial stability of the HAST chamber as measured by the Wormwood system. HAST chamber was set at 130C and 85% RH.

The design of the system was outlined including a description of the efforts necessary to allow measurement of temperature under the 130C, 85% RH HAST conditions encountered. The system was then used to characterize the HAST chambers found in Sandia National Laboratories Dept. 1718.

Data demonstrate the excellent temporal stability (±5C) over many hours is possible in the system and spatial variations occur with standard deviations on the order of ±10C. In conclusion, this design fulfills a specific niche requirement to operate under extreme conditions and report temperature data in HAST systems.

4. REFERENCES

[1] Hubert P Kelsey Ed., "*Webster's Ninth New Collegiate Dictionary*," Merriam-Webster Inc. Springfield, Massachusetts, (1985) pg. 1360.

[2] Quora, *What does Chernobyl mean in the Russian and Ukrainian languages?,* Sept. 12, 2015, https://www.quora.com/What-does-Chernobyl-mean-in-the-Russian-and-Ukrainian-languages.

[3] A. Mavinkurve, J.L.M. Llacer Martinez, M. van Soestbergen, J.J.M. Zaal, "Moisture Absorption by Molding Compounds Under Extreme Conditions: Impact on Accelerated Reliability Tests," *Microelectronics Reliability* 64 (2016) 254–258.

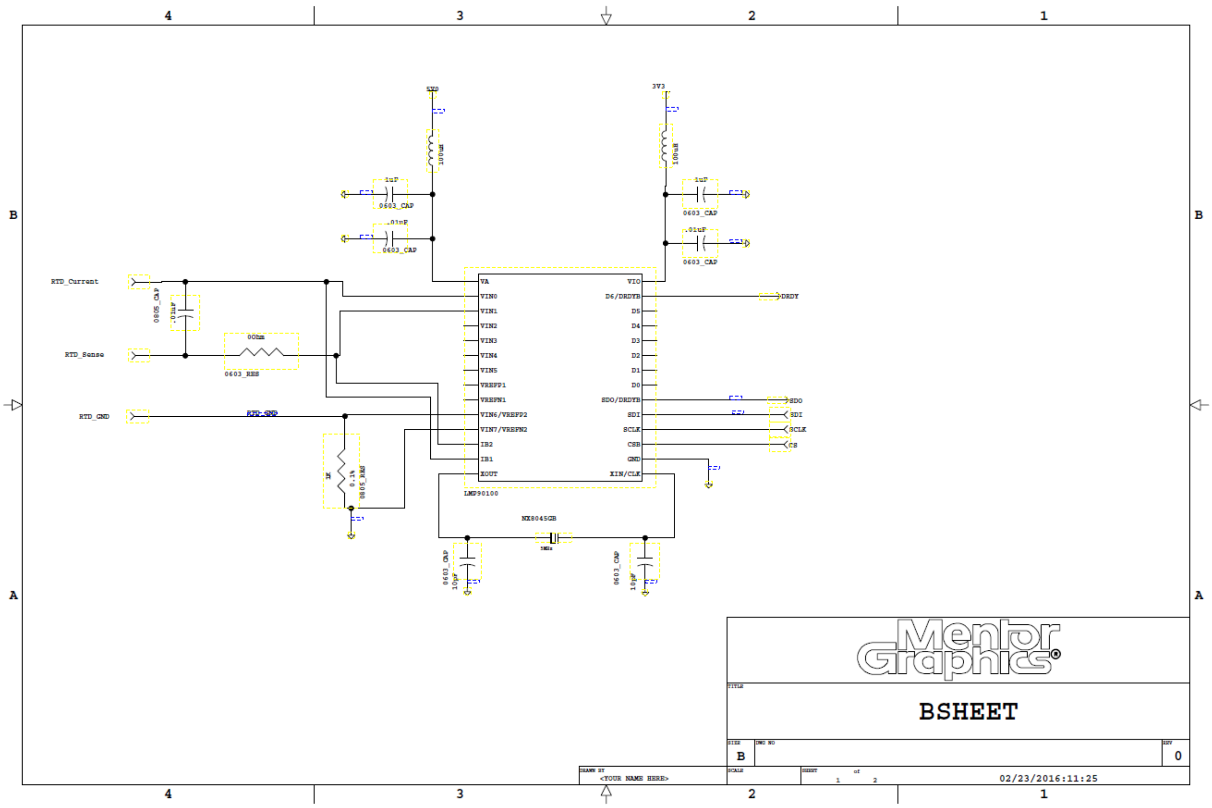
[4] C. Pei, G. Fu, B. Wan, L. Zhong, M. Jia, "Failure Analysis of Plastic Encapsulated Circuits After Autoclaving", *Engineering Failure Analysis* 66 (2016) 421–431.

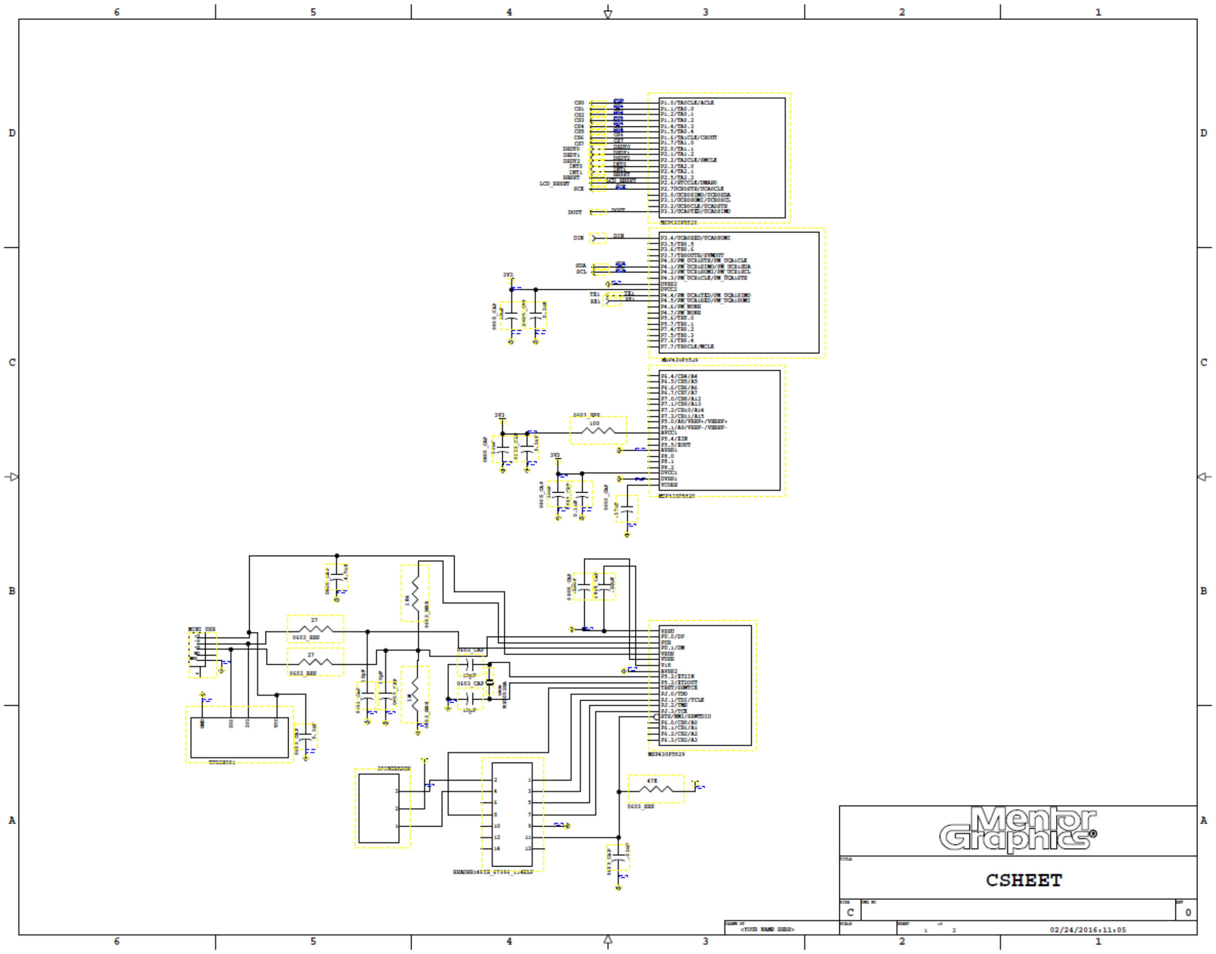
[5] F. R. Gantmacher, "The Theory of Matrices," Chelsea Publishing Co. New York, New York, (1977) pp. 66-68.

[6] M. G. Natrella, "Experimental Statistics", National Bureau of Standards Handbook 91, (1966) p. 1-10 -1-14.

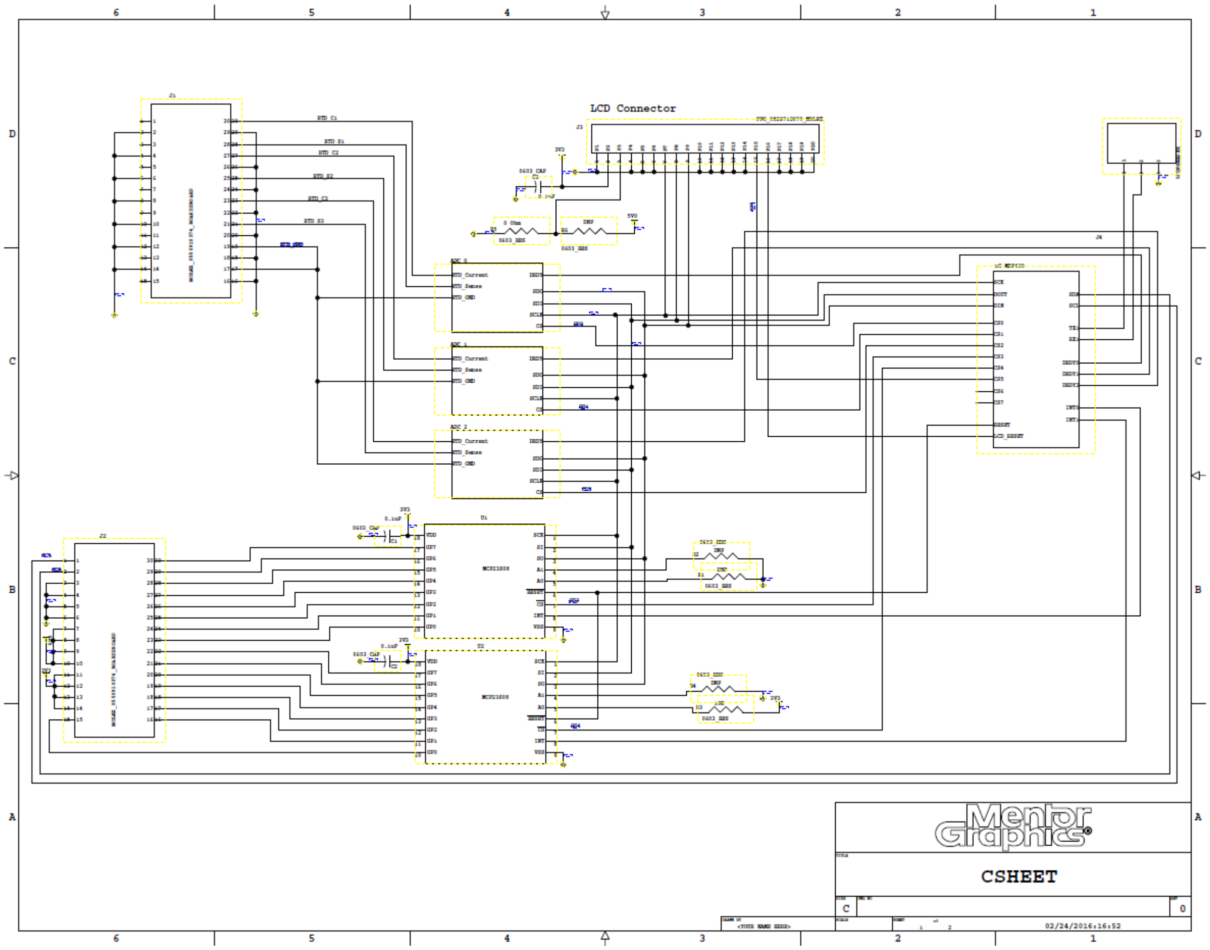
APPENDIX A: SYSTEM SCHEMATICS

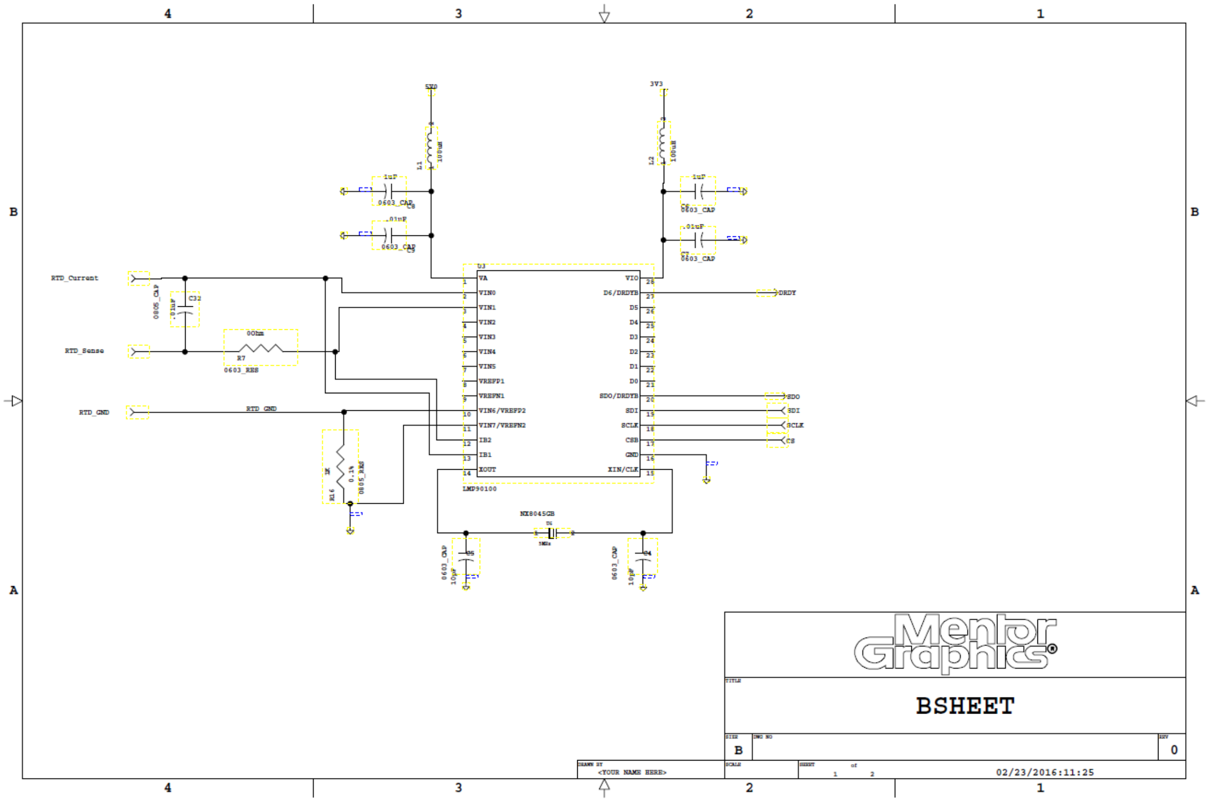
Wormwood Logic Board:



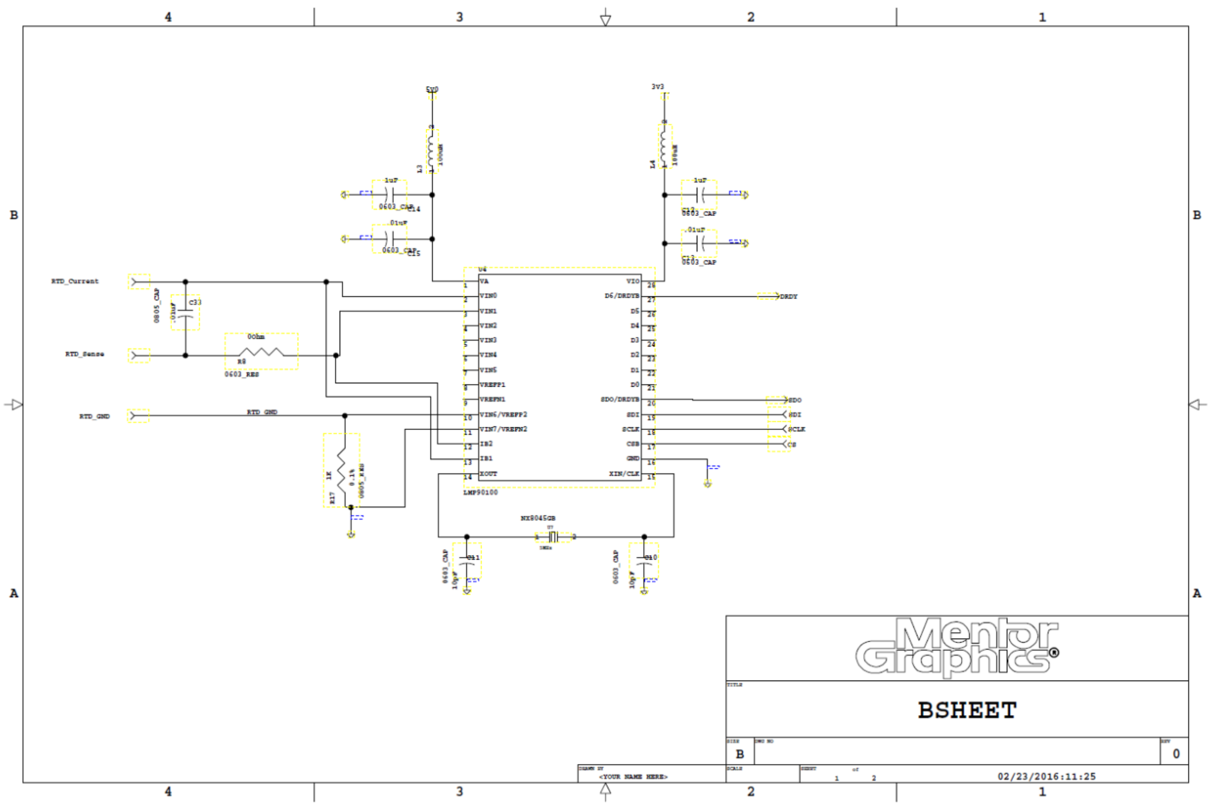


CSHEET
02/24/2016:11:05

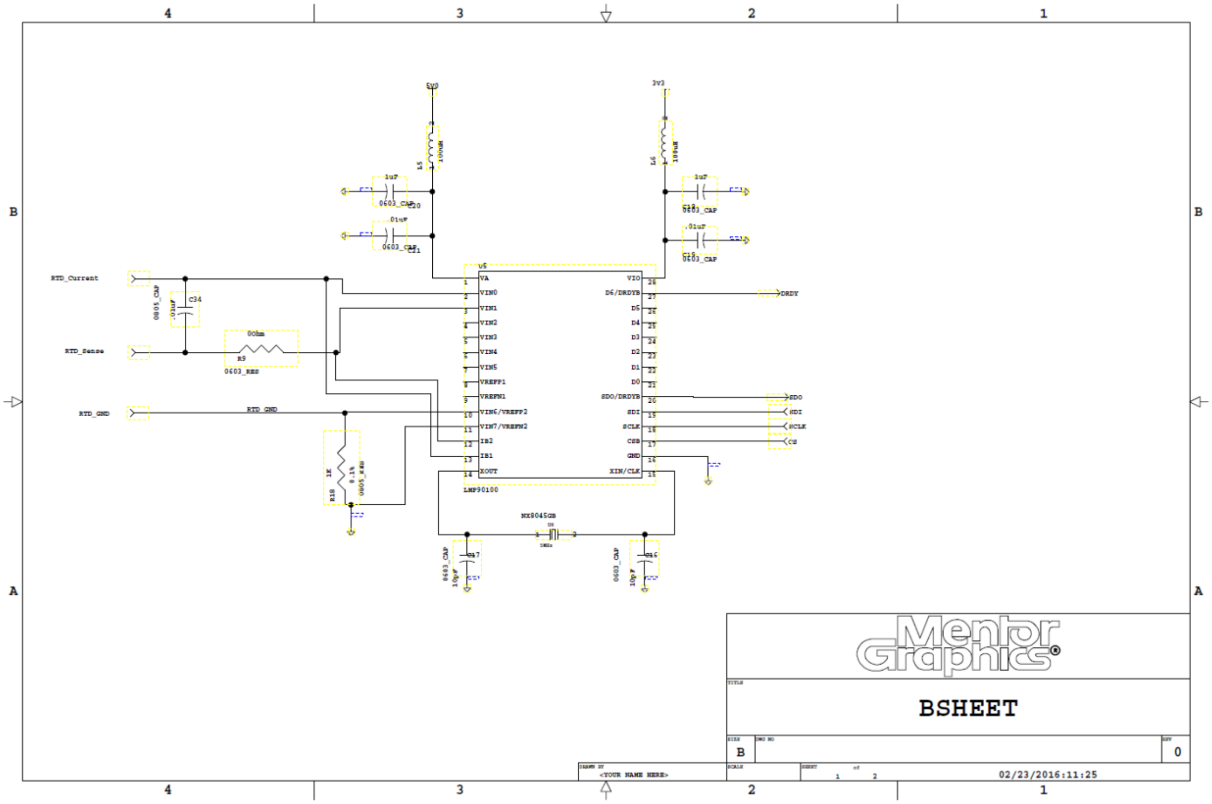

LCD Connector
ADC 0
ADC 1
ADC 2
MCP23S08
CSHEET
Mentor Graphics
02/24/2016:16:52

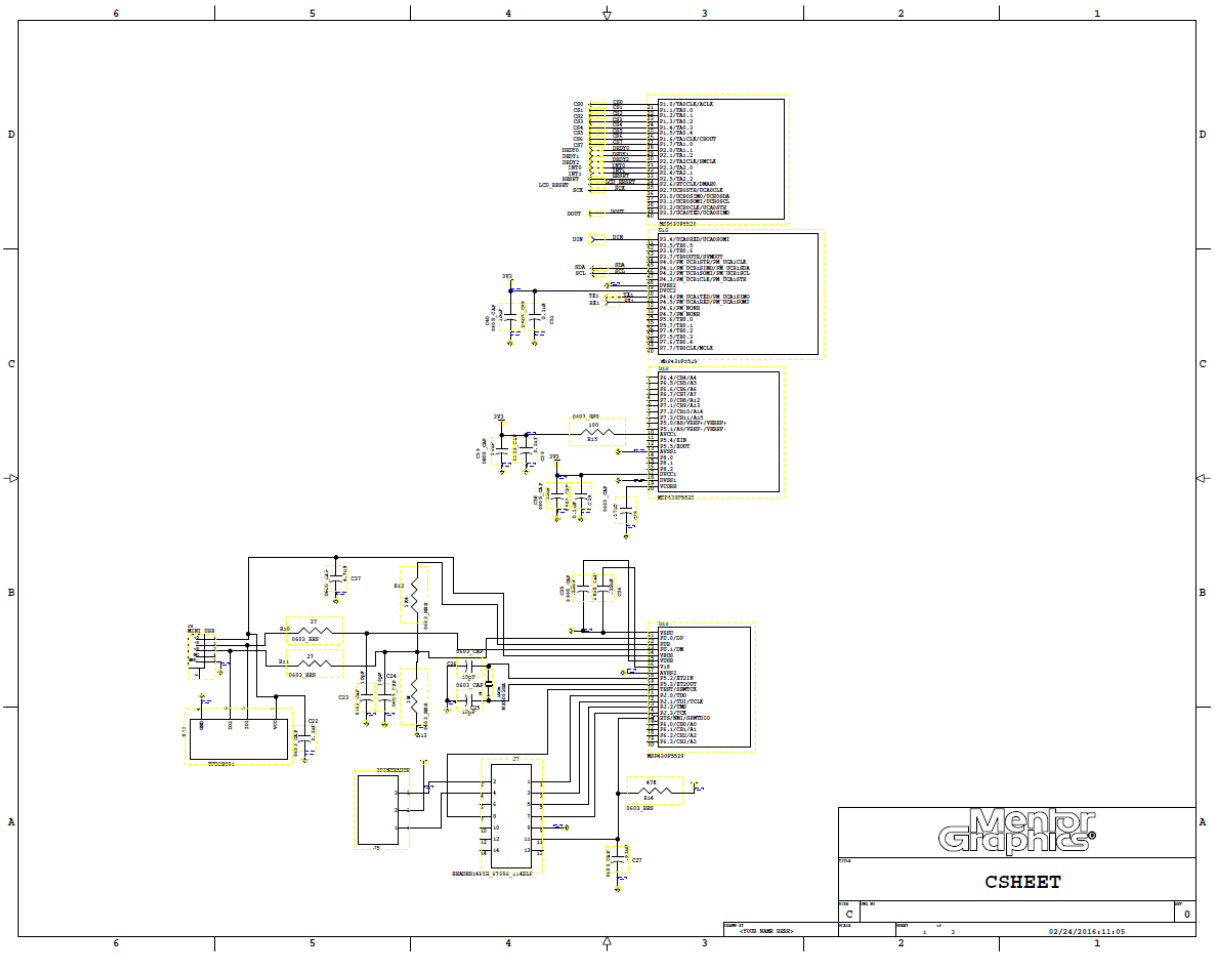


RTD_Current
RTD_Sense
RTD_GND
LMP90100
NX8045GB
DRDY
SDO
SDI
SCLK
CS
BSHEET
02/23/2016:11:25

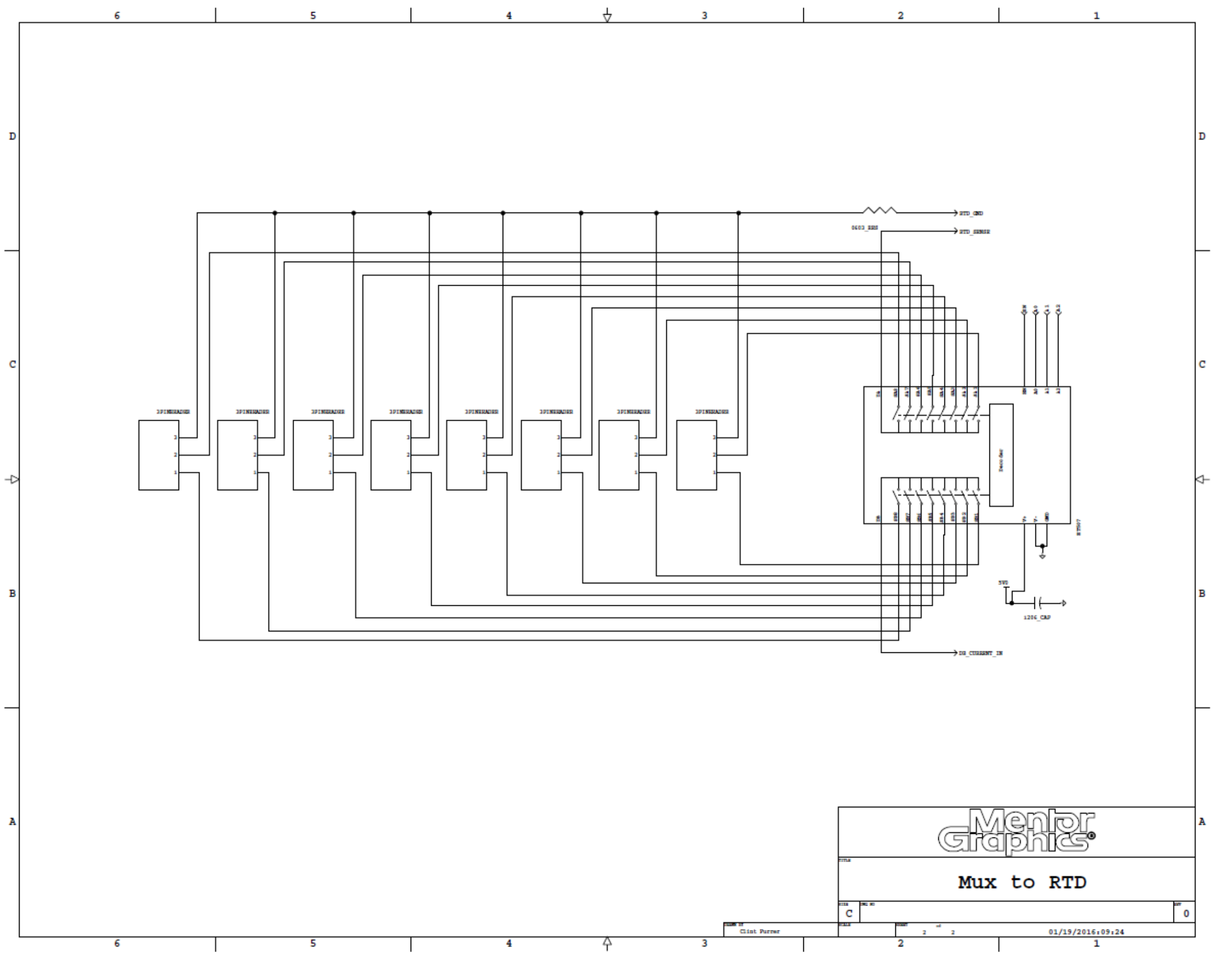


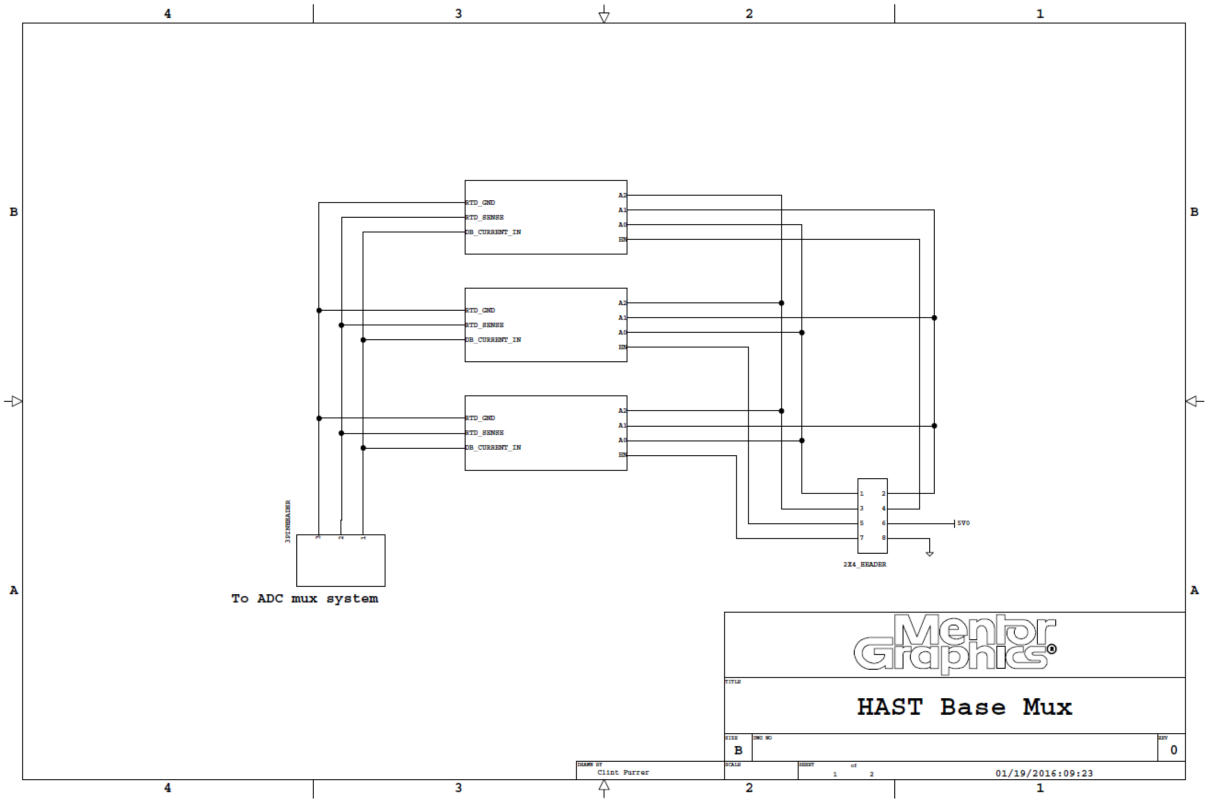
RTD_Current
RTD_Sense
RTD_GND
LMP90100
NX8045GB
DRDY
SDO
SDI
SCLK
CS
BSHEET
02/23/2016:11:25


BSHEET

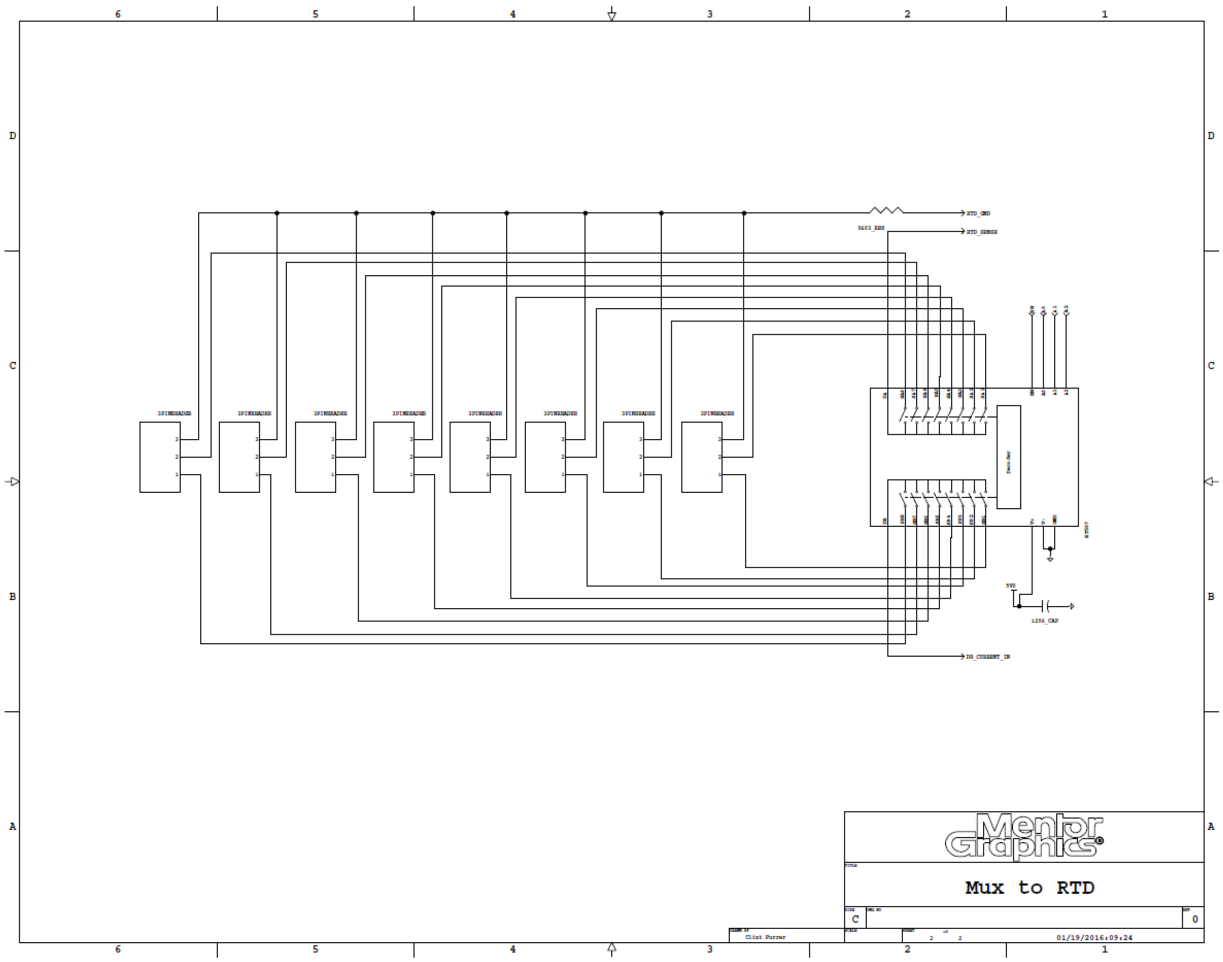

CSHEET
02/24/2016:11:05

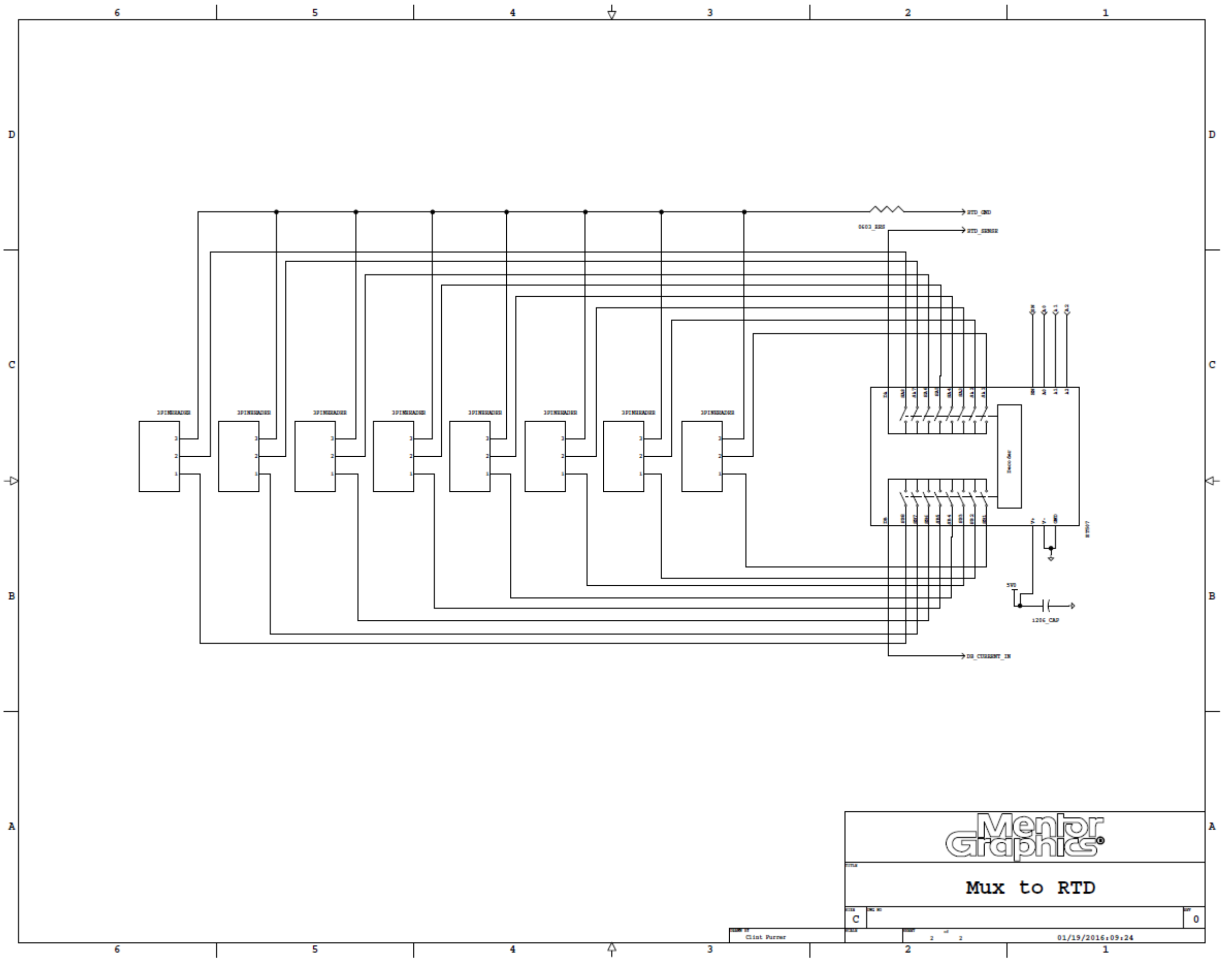
Wormwood Temperature Card:

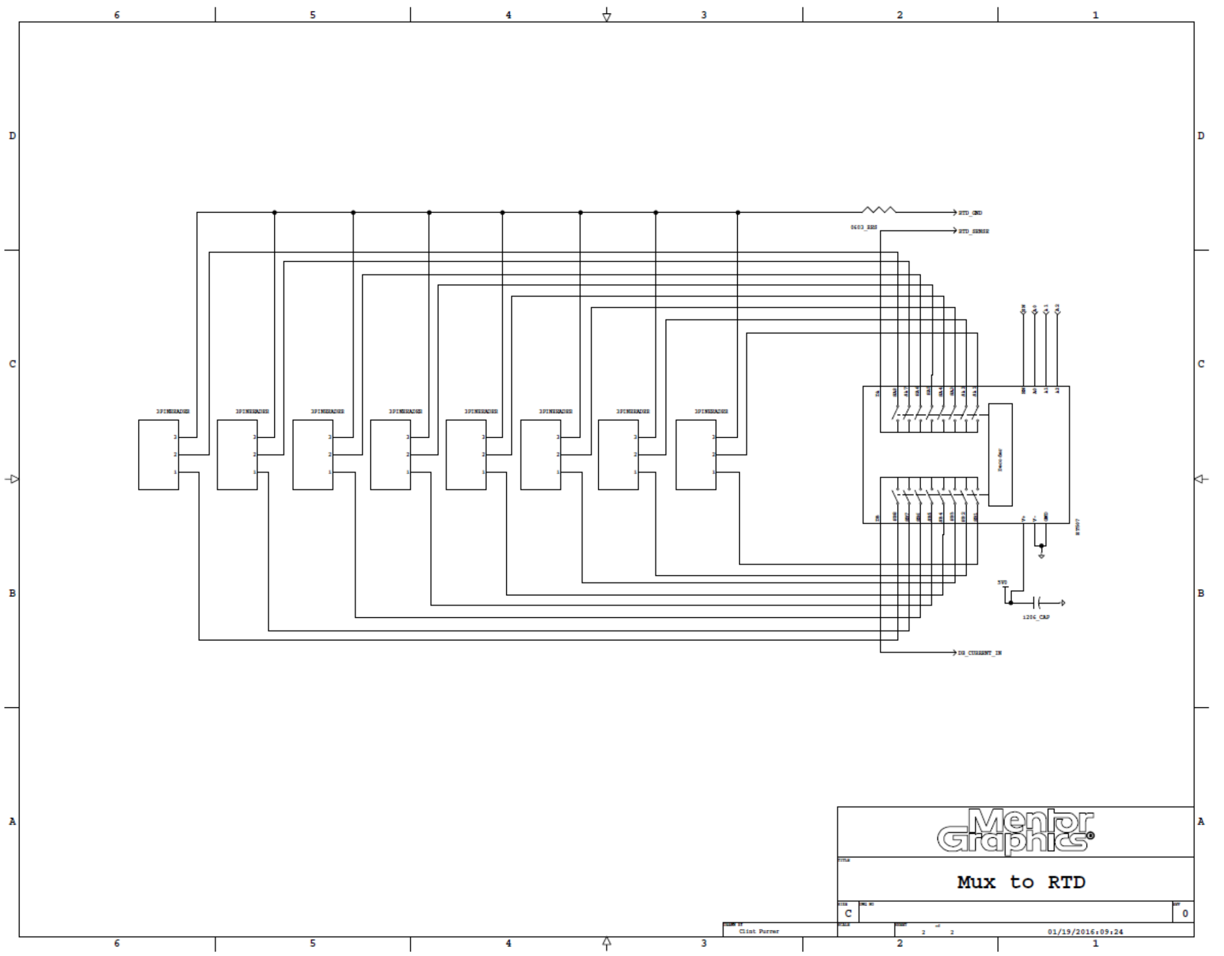




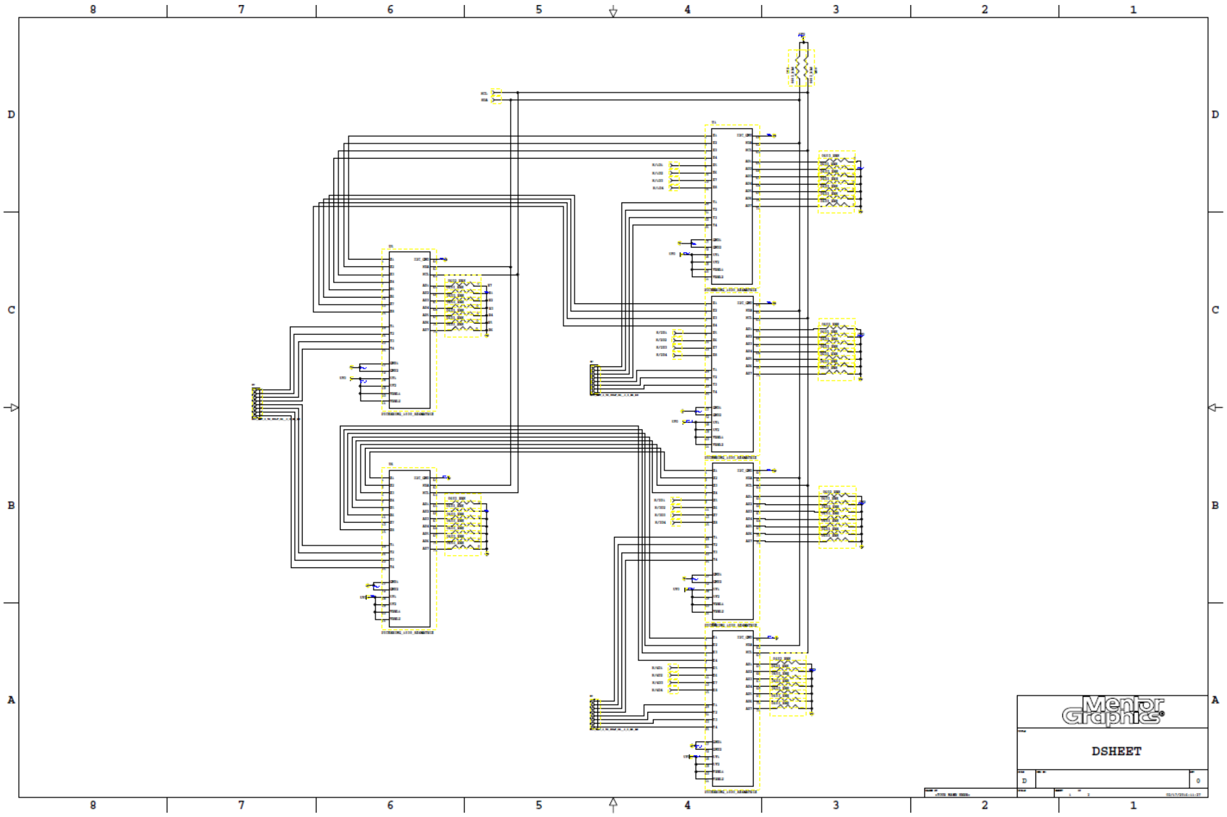
RTD_GND
RTD_SENSE
DB_CURRENT_IN
A2
A1
A0
EN
3PINHEADER
To ADC mux system
2X4_HEADER
5V0
Mentor Graphics
HAST Base Mux
B
0
Clint Furrer
01/19/2016:09:23

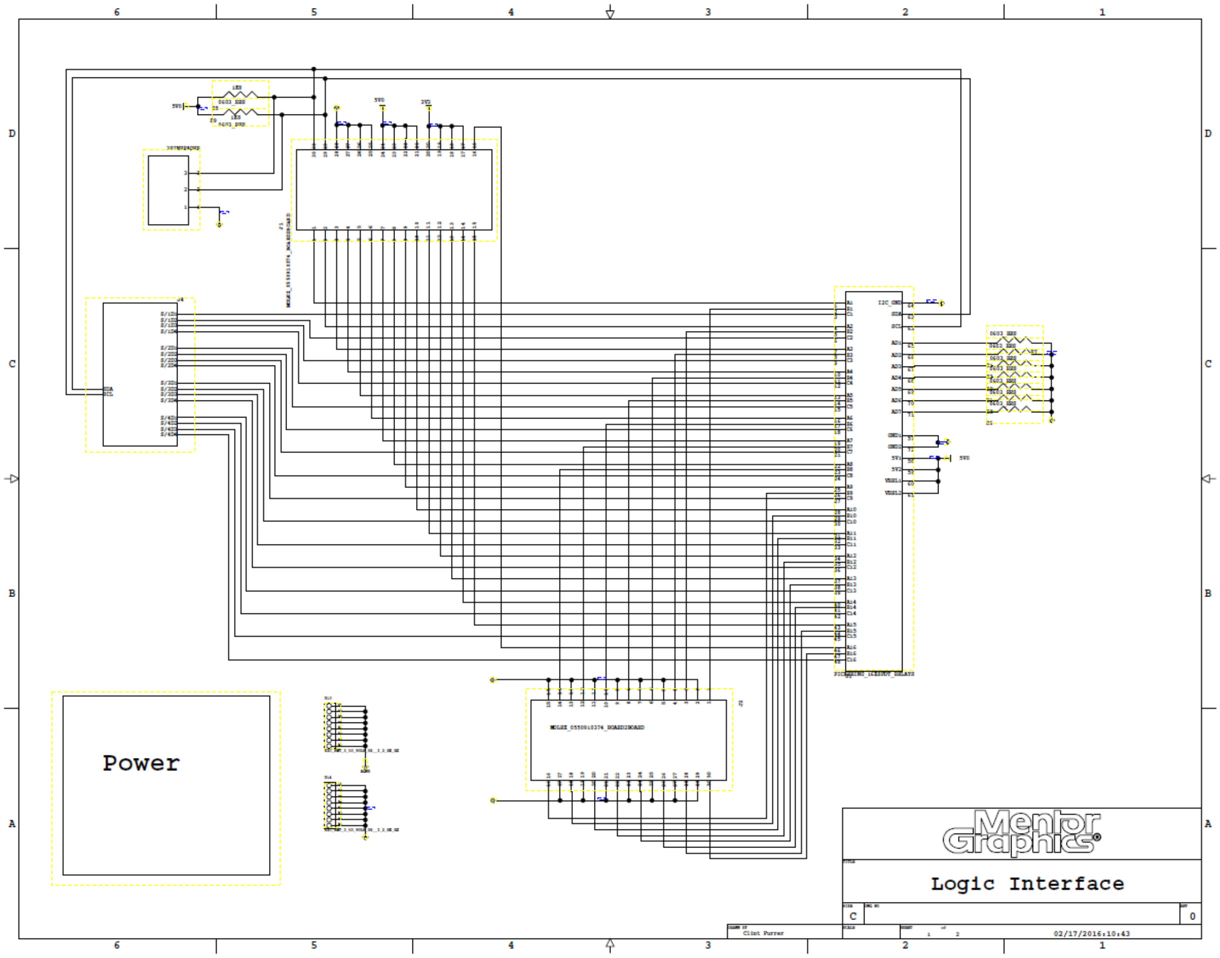

Mux to RTD
C
0
Clint Purrer
2
2
01/19/2016:09:24

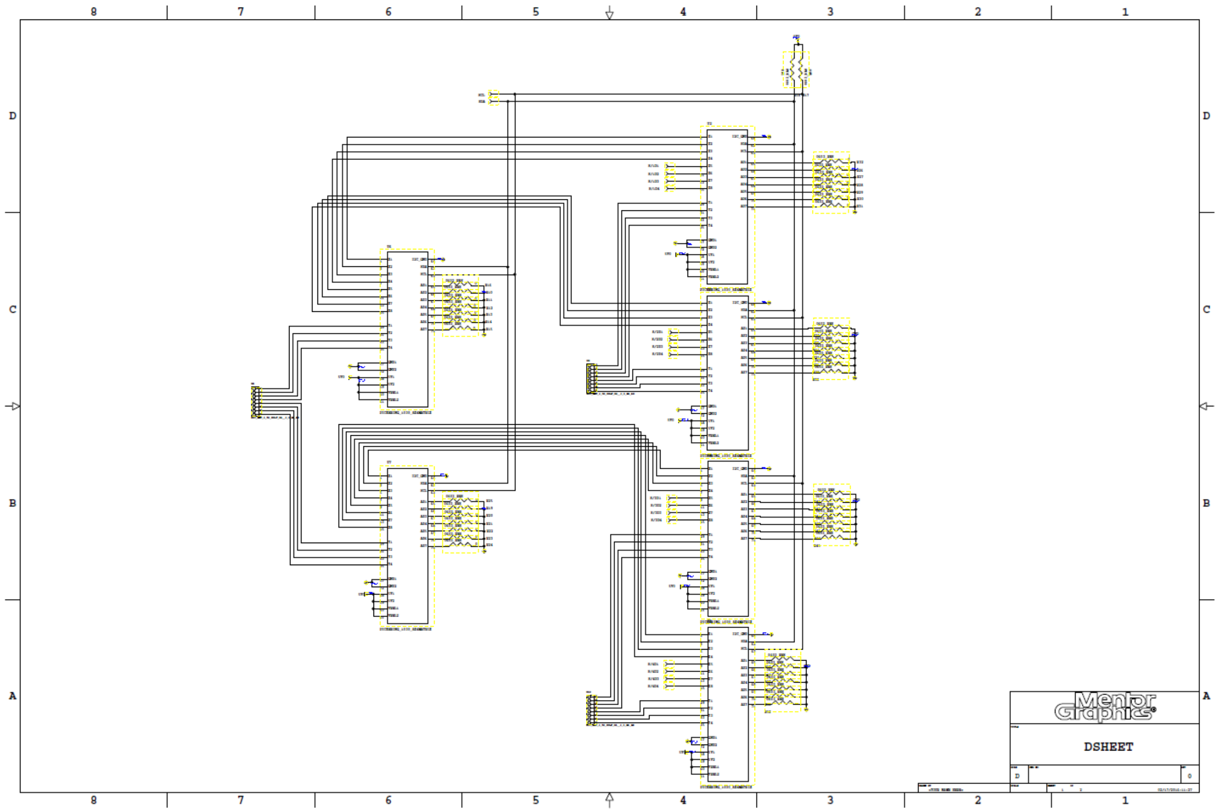

RTD_GND
0603_RES
RTD_SENSE
5V0
1206_CAP
DS_CURRENT_IN
Mux to RTD
C
0
Clint Furrer
01/19/2016:09:24

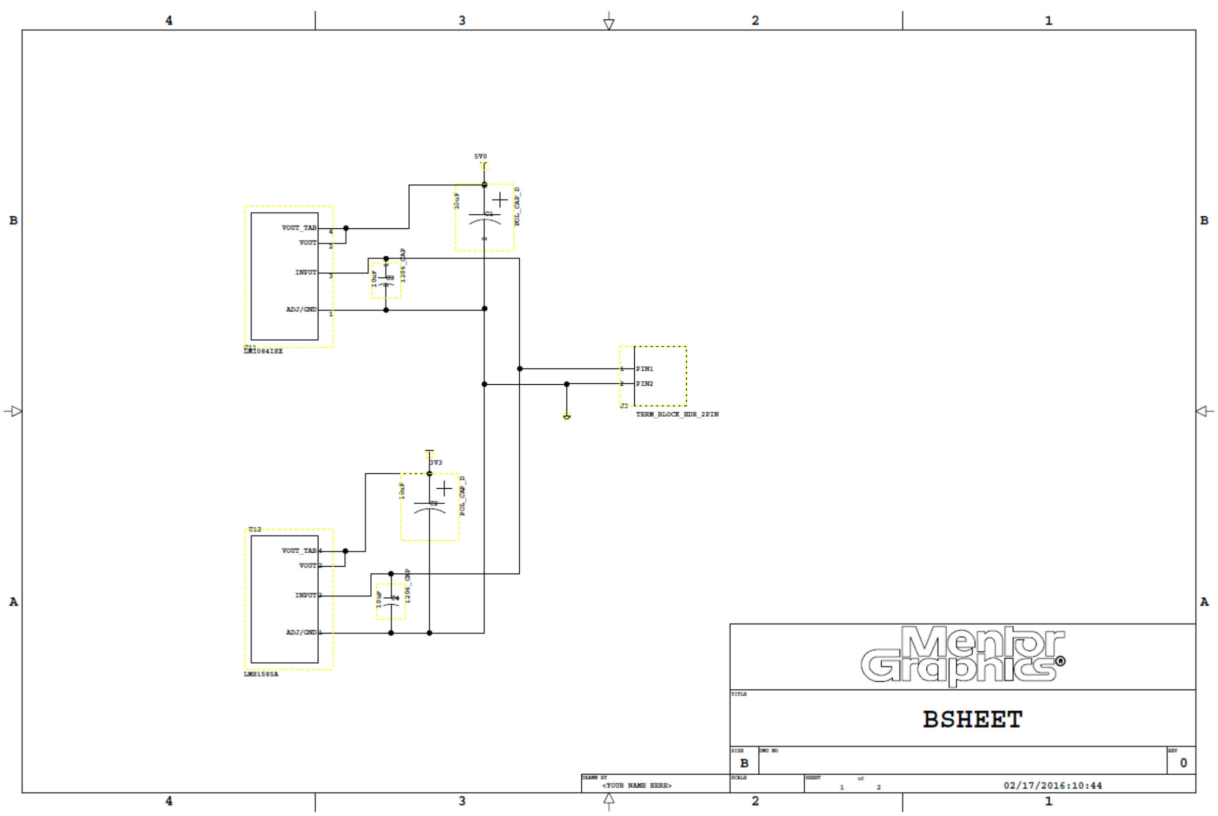

RTD_GND
RTD_SENSE
DR_CURRENT_IN
0603_RES
1206_CAP
3PINHEADER
Mux to RTD
Clint Furrer
01/19/2016:09:24

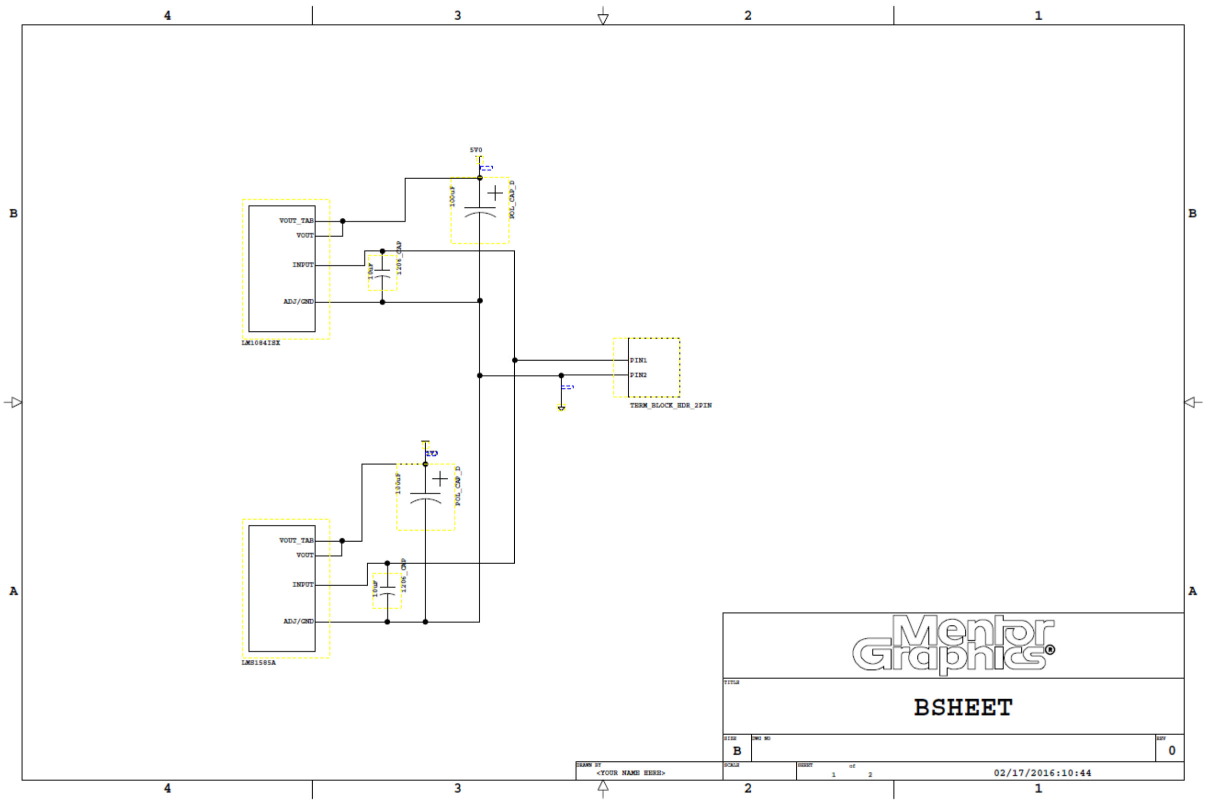
Wormwood Switch Matrix:




Power
Logic Interface
Clint Furrer
02/17/2016:10:43


DSHEET


5V0
10uF
C1
POL_CAP_D
VOUT_TAB
VOUT
INPUT
ADJ/GND
10uF
1206_CAP
PIN1
PIN2
J3
TERM_BLOCK_HDR_2PIN
3V3
10uF
C2
POL_CAP_D
U12
VOUT_TAB
VOUT
INPUT
ADJ/GND
10uF
1206_CAP
LMS1585A
TITLE
BSHEET
SIZE
B
DWG NO
REV
0
DRAWN BY
<YOUR NAME HERE>
SCALE
SHEET
1 of 2
02/17/2016:10:44



5V0
VOUT_TAB
VOUT
INPUT
ADJ/GND
LM1084ISX
100uF
POL_CAP_D
10uF
1206_CAP
PIN1
PIN2
TERM_BLOCK_HDR_2PIN
LMS1585A
BSHEET
B
0
<YOUR NAME HERE>
02/17/2016:10:44

APPENDIX B: SYSTEM SOFTWARE

Python code for data acquisition that operated the Wormwood system.

```
'''
Created on Mar 28, 2016

@author: ctfurre
'''

import serial
import struct
import time
#import traceback
#import os
import threading, msvcrt
import sys
from os.path import expanduser

def readInput(caption, default, timeout = 5):
    class KeyboardThread(threading.Thread):
        def run(self):
            self.timedout = False
            self.input = ''
            while True:
                if msvcrt.kbhit():
                    chrr = msvcrt.getche()
                    if ord(chrr) == 13:
                        break
                    elif ord(chrr) >= 32:
                        self.input += chrr
                if len(self.input) == 0 and self.timedout:
                    break


    sys.stdout.write('%s(%s):'%(caption, default));
    result = default
    it = KeyboardThread()
    it.start()
    it.join(timeout)
    it.timedout = True
    if len(it.input) > 0:
        # wait for rest of input
```

```
        it.join()
        result = it.input
    print ''  # needed to move to next line
    return result


port = raw_input('What PORT do you want to connect to?: ')
HS2M = serial.Serial()

HS2M.port = int(port)
HS2M.setBaudrate(38400)
HS2M.setParity(serial.PARITY_NONE)
HS2M.setStopbits(serial.STOPBITS_ONE)
HS2M.setByteSize(serial.EIGHTBITS)
HS2M.writeTimeout = 0

try:
    HS2M.open()
except Exception, err:
    print err

home = expanduser("~")
path = raw_input('Enter your filename example(Mydata.txt): ')

datafile = open(home+'\\'+path,'w')

debugfile = open(home+'\WormwoodDebug.txt','w')

rawdatafile = open(home+'\\'+'raw'+path,'w')
rawDetlaTfile = open(home+'\\'+'rawDelta'+path,'w')

output = struct.pack('ccccBBBBBB', 'S','W','C','L',0,0,0,0,0,0)
HS2M.write(output)

next = raw_input('sent switch close command press enter?: ')
message = HS2M.read(HS2M.inWaiting())
print message

output = struct.pack('ccccBBBBBB', 'C','O','N','D',0,0,0,0,0,0)
HS2M.write(output)

next = raw_input('sent digital IO command press enter?: ')
message = HS2M.read(HS2M.inWaiting())
print message

output = struct.pack('ccccBBBBBB', 'S','Y','U','P',0,0,0,0,0,0)
```

```
HS2M.write(output)

next = raw_input('sent Sys up command press enter?: ')
message = HS2M.read(HS2M.inWaiting())
print message

setpoint = raw_input('Enter the set point?: ')
if setpoint == '':
    setpoint = raw_input('PLEASE enter a set point!: ')

next = raw_input('Start the data scan?: ')

zeropointList = []
TempdataList = []
q = range(72)
TempdataList = q
RawDataList = q
DataCount = 0
TempTrigger = 0
NOErrorFlag = 1

if next == 'YES':
    testloop = 1
    try:
        while testloop:
            ans = readInput('Press ctrl-c to stop logging you have 10sec ', 'n',10)

            output = struct.pack('ccccBBBBBB', 'S','C','A','N',0,0,0,0,0,0)
            HS2M.write(output)
            print 'Wormwood scanning'
            time.sleep(40)#HS2M is sampling and processing the RTD measurements
            RTDdata = HS2M.read(HS2M.inWaiting())

            RTD = RTDdata.split('DATA')#split the samples apart from each other
            #print RTD
            del RTD[len(RTD)-1]
            #RTD = RTD[:24]
            #print RTD
            #datafile.write(time.ctime(time.time())+',\r')

            for y in RTD:
                #print y
                try:
                    unpackedData = struct.unpack('!I',('\x00'+y))
                    #print unpackedData
                    temp = float(unpackedData[0])
```

```
        #print temp
        degreeC = 2.5699*((temp*4000)/16777216)-257.01
        RawDataList[DataCount] = temp
        TempdataList[DataCount] = degreeC
        DataCount = DataCount + 1

        #print degreeC
        #datafile.write(str(degreeC))
        #datafile.write(',')
        #datafile.write('\r')
    except:
        print "!!!DATA STREAM ERROR!!!"
        print "data point got skipped but I will keep asking for data..."
        debugfile.write(time.ctime(time.time())+',\r')
        debugfile.write(str(RTDdata)+',\r')
        debugfile.write(str(RTD)+',\r')
        NOErrorFlag = 0
print "Current reading of BEST RTD 12 is:"

DataCount = 0
if NOErrorFlag:
    print TempdataList[11]
    if TempTrigger == 0:
        if TempdataList[11] >= int(setpoint):
            print"Temperature triggered"
            zeropointList = list(TempdataList)
            rawZeropointList = list(RawDataList)
            datafile.write(time.ctime(time.time())+',\r')
            for x in range(len(TempdataList)):
                DeltaT = TempdataList[x] - zeropointList[x]
                rawDeltaT = RawDataList[x] - rawZeropointList[x]
                #print DeltaT
                rawDetlaTfile.write(str(rawDeltaT))
                rawdatafile.write(str(RawDataList[x]))
                datafile.write(str(DeltaT))
                datafile.write(',')
                rawdatafile.write(',')
                rawDetlaTfile.write(',')
            datafile.write('\r')
            rawdatafile.write('\r')
            rawDetlaTfile.write('\r')
            TempTrigger = 1

    else:
        datafile.write(time.ctime(time.time())+',\r')
        for x in range(len(TempdataList)):
```

```
                    DeltaT = TempdataList[x] - zeropointList[x]
                    rawDeltaT = RawDataList[x] - rawZeropointList[x]
                    #print "zero point array"
                    #print zeropointList
                    #print "TempdataList"
                    #print TempdataList
                    #print DeltaT
                    print TempdataList[x]
                    rawDetlaTfile.write(str(rawDeltaT))
                    rawdatafile.write(str(RawDataList[x]))
                    datafile.write(str(DeltaT))
                    datafile.write(',')
                    rawdatafile.write(',')
                    rawDetlaTfile.write(',')
                datafile.write('\r')
                rawdatafile.write('\r')
                rawDetlaTfile.write('\r')
        else:
            print"There was Errors in the data stream No problem will try again. Catch you on the
next loop :P"
            NOErrorFlag = 1


        #print 'got data'
        #time.sleep(30)
            #print unpackedData
      #for x in unpackedData:
    except KeyboardInterrupt:
      print 'Stopped!'
      output = struct.pack('ccccBBBBBB', 'S','W','O','P',0,0,0,0,0,0)
      HS2M.write(output)
      testloop = 0


print 'Bye'
HS2M.close()
datafile.close()
debugfile.close()
rawdatafile.close()
rawDetlaTfile.close()
```

APPENDIX C: ANIMATION SOFTWARE

MATLab Code for Animation of the Wormwood Data

```
%% HASTtempanalysis
%%
%% Program by N. B. Pfeifer
%% Sandia National Laboratories
%% Nano and Micro Sensor, Dept. 8634
%% Program Animates Wormwood temperature data from HAST systems
%% into 2-D temperature “topographic” maps as a function of time
%%
%Program takes inputs of temperature and sensor locations in cartesian
%coordinates and treats temperature like a third axis indicating the
%difference on the Y-Z plane

%% Inputs
%Excel file name with temp data must be in same folder as m-file
Tfile='HAST_Stability_130C_102716_Noise_Removed.xlsx';
%Name of sheets
Tfilesheet='HAST_Noise_Removed';
Tray1='Tray1';
Tray2='Tray2';
Tray3='Tray3';
ydim=1:.05:4;
zdim=1:.05:10;
min_c=20;
max_c=150;


%% Import Data
%Temperature data
Temp=xlsread(Tfile,Tfilesheet);

%Sensor data
tray1sensors=xlsread(Tfile,Tray1);
tray2sensors=xlsread(Tfile,Tray2);
tray3sensors=xlsread(Tfile,Tray3);

%% Generate movie
%determine the number of sensors on each tray
[nums1,~]=size(tray1sensors);
[nums2,~]=size(tray2sensors);
[nums3,~]=size(tray3sensors);
%determine the number of times analyzed
[duration,~]=size(Temp);
%Each loop creates a contour plot which is saved as a frame of the movie


%kbp added following lines
```

```
vidObj1 = VideoWriter('HASTtest_321.avi');
vidObj1.Quality = 100;
vidObj1.FrameRate = 10;
open(vidObj1);


for jj=2:duration
%initalize value colection matrixes
seny1=[];
senz1=[];
sensordata1=[];
%choose time of analysis
Time=jj;
%Generate vectors with spacial and thermal data for contour plot

%Tray1
for ii=1:nums1
    %sensor number
    sennum=tray1sensors(ii,1);
    %collect sensor location in y-direction and add to list
    seny1=[seny1;tray1sensors(ii,2)];
    %collect sensor location in z-direction and add to list
    senz1=[senz1;tray1sensors(ii,3)];
    %collect thermal data at correct sensor and time and add to list
    sensordata1=[sensordata1;Temp(Time,sennum)];
end
%Repeat above steps for other trays
seny2=[];
senz2=[];
sensordata2=[];
for ii=1:nums2
    sennum=tray2sensors(ii,1);
    seny2=[seny2;tray2sensors(ii,2)];
    senz2=[senz2;tray2sensors(ii,3)];
    sensordata2=[sensordata2;Temp(Time,sennum)];
end
seny3=[];
senz3=[];
sensordata3=[];
for ii=1:nums3
    sennum=tray3sensors(ii,1);
    seny3=[seny3;tray3sensors(ii,2)];
    senz3=[senz3;tray3sensors(ii,3)];
    sensordata3=[sensordata3;Temp(Time,sennum)];
end

%Create Griddata
%Create an evenly spaced 3-d mesh grid to where one demenstion is temp
[Y1,Z1]=meshgrid(ydim,zdim);
[Y2,Z2]=meshgrid(ydim,zdim);
[Y3,Z3]=meshgrid(ydim,zdim);
%linearly interpolate to fill in missing values for plot
 int1=griddata(senz1,seny1,sensordata1,Z1,Y1);
 int2=griddata(senz2,seny2,sensordata2,Z2,Y2);
 int3=griddata(senz3,seny3,sensordata3,Z3,Y3);
```

```
%% Generate contour plots
%create figure for tray1
figure(1)
%Define subplot position for tray1
subplot(3,1,1)
%generate contur plot
contourf(Z1,Y1,int1)
%add color bar
caxis([min_c,max_c])
colorbar('eastoutside')
set(gca,'Xtick',[])
title('Tray 1','fontsize',8)
subplot(3,1,2)
%repeat for trays2-3
% figure(2)
contourf(Z2,Y2,int2)
caxis([min_c,max_c])
colorbar('eastoutside')
set(gca,'Xtick',[])
title('Tray 2','fontsize',8)
% xlabel('Z direction')
ylabel('Y Direction','fontsize',12)
%writeVideo(vidObj2,getframe(2));
%F2(jj)=getframe(2);
% close(2)
% figure(3)
subplot(3,1,3)
contourf(Z3,Y3,int3)
caxis([min_c,max_c])
colorbar('eastoutside')
title('Tray 3','fontsize',8)
xlabel('Z Direction','fontsize',12)
writeVideo(vidObj1,getframe(1));
%close figure
close(1)


end
close(vidObj1);
```

DISTRIBUTION

1	MS0352	Andre Claudet	1718
1	MS0352	Stephen E. Garrett	1718
1	MS0352	Clint T. Furrer III	1718
1	MS0352	Paul A. Sandoval	1718
1	MS0892	Ronald Manginell	8634
1	MS0892	Matthew Moorman	8634
1	MS0892	Nathaniel B. Pfeifer	8634
1	MS1072	David Brumit	1754
1	MS1072	Aaron Strasburg	1754
1	MS1425	Steven A. Casalnuovo	8630
1	MS1425	Kent B. Pfeifer	8634
1	MS0899	Technical Library	9536 (electronic copy)
1	MS0161	Legal Technology Transfer Center	11500

Sandia National Laboratories